# IEC 61158-6-3

Edition 2.0 2010-08

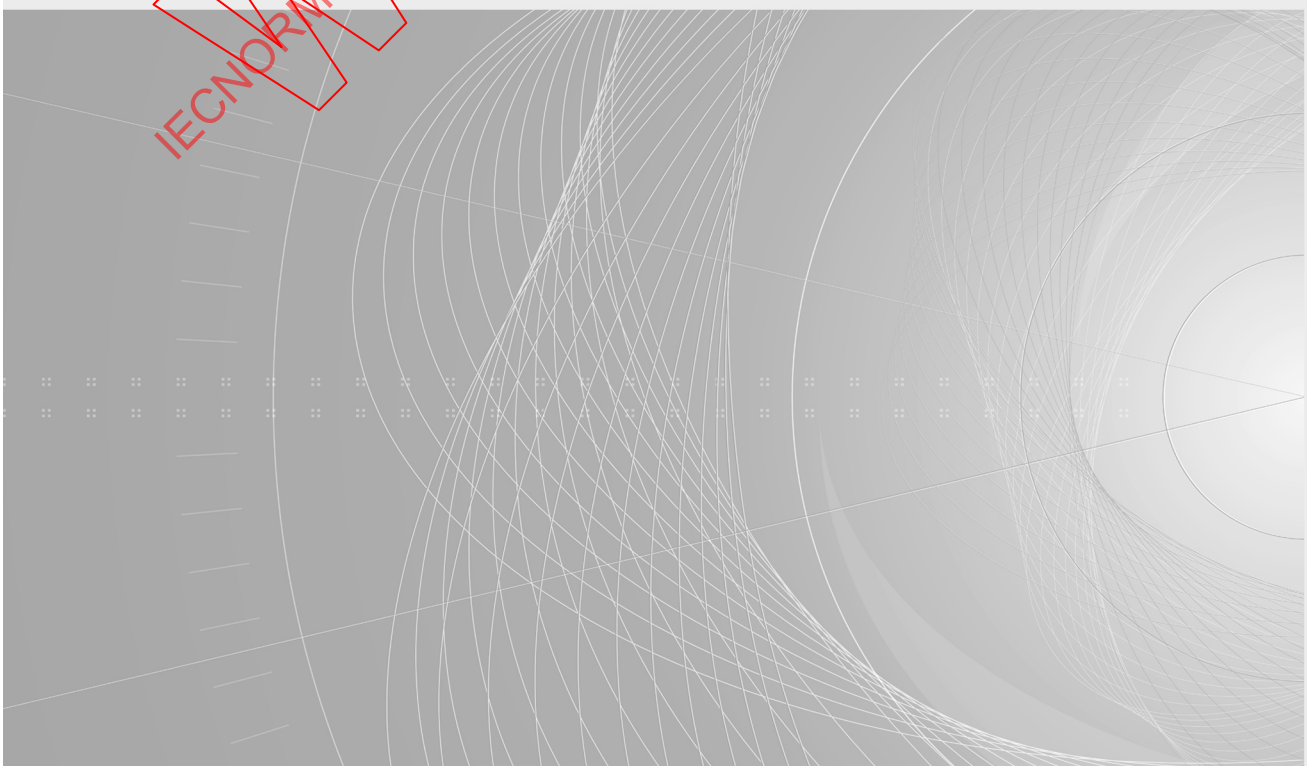
# INTERNATIONAL STANDARD

# NORME INTERNATIONALE

## Industrial communication networks – Fieldbus specifications – Part 6-3: Application layer protocol specification – Type 3 elements

## Réseaux de communication industriels – Spécifications des bus de terrain – Partie 6-3: Spécification de protocole de la couche application – Eléments de Type 3

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn



**THIS PUBLICATION IS COPYRIGHT PROTECTED**
**Copyright © 2010 IEC, Geneva, Switzerland**

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester.
If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de la CEI ou du Comité national de la CEI du pays du demandeur.
Si vous avez des questions sur le copyright de la CEI ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de la CEI de votre pays de résidence.

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

## About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

## About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

**Useful links:**

IEC publications search - www.iec.ch/searchpub

The advanced search enables you to find IEC publications by a variety of criteria (reference number, text, technical committee,…).
It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available on-line and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary (IEV) on-line.

Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: csc@iec.ch.

## A propos de la CEI

La Commission Electrotechnique Internationale (CEI) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

## A propos des publications CEI

Le contenu technique des publications de la CEI est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

**Liens utiles:**

Recherche de publications CEI - www.iec.ch/searchpub

La recherche avancée vous permet de trouver des publications CEI en utilisant différents critères (numéro de référence, texte, comité d'études,…).
Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

Just Published CEI - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications de la CEI. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org

Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 30 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (VEI) en ligne.

Service Clients - webstore.iec.ch/csc

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: csc@iec.ch.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

IEC 61158-6-3

Edition 2.0 2010-08

# INTERNATIONAL STANDARD

# NORME INTERNATIONALE

## Industrial communication networks – Fieldbus specifications – Part 6-3: Application layer protocol specification – Type 3 elements

## Réseaux de communication industriels – Spécifications des bus de terrain – Partie 6-3: Spécification de protocole de la couche application – Eléments de Type 3

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

INTERNATIONAL ELECTROTECHNICAL COMMISSION

COMMISSION ELECTROTECHNIQUE INTERNATIONALE

PRICE CODE XH
CODE PRIX

ICS 25.040.40; 35.100.70; 35.110

ISBN 978-2-83220-128-2

**Warning! Make sure that you obtained this publication from an authorized distributor.**
**Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.**

® Registered trademark of the International Electrotechnical Commission
Marque déposée de la Commission Electrotechnique Internationale

# CONTENTS

FOREWORD ..... 8
INTRODUCTION ..... 10
1 Scope ..... 12
1.1 General ..... 12
1.2 Specifications ..... 13
1.3 Conformance ..... 13
2 Normative references ..... 13
3 Terms, definitions, abbreviations, symbols and conventions ..... 14
3.1 Referenced terms and definitions ..... 14
3.2 Additional definitions ..... 15
3.3 Abbreviations and symbols ..... 18
3.4 Conventions ..... 19
3.5 Conventions used in state machines ..... 21
4 FAL syntax description ..... 24
4.1 APDU abstract syntax ..... 24
4.2 Data types ..... 28
5 Transfer syntax ..... 30
5.1 Coding of basic data types ..... 30
5.2 Coding section related to data exchange PDUs ..... 33
5.3 Coding section related to slave diagnosis PDUs ..... 33
5.4 Coding section related to parameterisation PDU ..... 44
5.5 Coding section related to configurationPDUs ..... 50
5.6 Coding section related to global control PDUs ..... 54
5.7 Coding section related to clock-value-PDUs ..... 55
5.8 Coding section related to function identification and errors ..... 56
5.9 Coding section related to master diagnosis PDU ..... 60
5.10 Coding section related to upload/download/act para PDUs ..... 62
5.11 Coding section related to the bus parameter set ..... 64
5.12 Coding section related to the slave parameter set ..... 66
5.13 Coding section related to statistic counters ..... 70
5.14 Coding section related to set slave address PDU ..... 70
5.15 Coding section related to initiate/abort PDUs ..... 70
5.16 Coding section related to read/write/data transport PDUs ..... 74
5.17 Coding section related to load region and function invocation PDUs ..... 74
5.18 Examples of diagnosis-RES-PDUs ..... 77
5.19 Example of Chk_Cfg-REQ-PDU ..... 79
5.20 Examples of Chk_Cfg-REQ-PDUs with DPV1 data types ..... 80
5.21 Example structure of the Data_Unit for Data_Exchange ..... 81
6 FAL protocol state machines ..... 83
6.1 Overall structure ..... 83
6.2 Assignment of state machines to devices ..... 84
6.3 Overview DP-slave ..... 85
6.4 Overview DP-master (class 1) ..... 87
6.5 Overview DP-master (class 2) ..... 88
6.6 Cyclic communication between DP-master (class 1) and DP-slave ..... 89
6.7 Acyclic communication between DP-master (class 2) and DP-master (class 1) ..... 90

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

6.8 Acyclic communication between DP-master (class 1) and DP-slave ..... 92
6.9 Application relationship monitoring ..... 94
7 AP-context state machine ..... 100
8 FAL service protocol machines (FSPMs) ..... 100
8.1 FSPMS ..... 100
8.2 FSPMM1 ..... 136
8.3 FSPMM2 ..... 172
9 Application relationship protocol machines (ARPMs) ..... 190
9.1 MSCY1S ..... 190
9.2 MSAC1S ..... 221
9.3 SSCY1S ..... 234
9.4 MSRM2S ..... 237
9.5 MSAC2S ..... 243
9.6 MSCS1S ..... 260
9.7 MSCY1M ..... 262
9.8 MSAL1M ..... 282
9.9 MSAC1M ..... 291
9.10 MMAC1 ..... 304
9.11 MSCS1M ..... 311
9.12 MSAC2M ..... 316
9.13 MMAC2 ..... 331
10 DLL mapping protocol machines (DMPMs) ..... 338
10.1 DMPMS ..... 338
10.2 DMPMM1 ..... 351
10.3 DMPMM2 ..... 367
11 Parameters for a DP-slave ..... 375
Bibliography ..... 376

Figure 1 – Common structure of specific fields ..... 20
Figure 2 – Coding of the data type BinaryDate ..... 31
Figure 3 – Encoding of TimeOfDay value ..... 31
Figure 4 – Encoding of Time Difference value ..... 32
Figure 5 – Encoding of Network Time value ..... 32
Figure 6 – Encoding of Network Time Difference value ..... 33
Figure 7 – Example Modul_Status_Array ..... 38
Figure 8 – Example of Ext_Diag_Data in case of DPV1 diagnosis format with alarm and status PDU ..... 78
Figure 9 – Example of Ext_Diag_Data in case of the basic diagnosis format ..... 79
Figure 10 – Example of a special identifier format ..... 79
Figure 11 – Example of a special identifier format with data types ..... 80
Figure 12 – Example of a special identifier format with data types ..... 80
Figure 13 – Example of a empty slot with data types ..... 81
Figure 14 – Example for multi-variable device with AI and DO function blocks ..... 81
Figure 15 – Identifiers (ID) ..... 82
Figure 16 – Identifier list ..... 82
Figure 17 – Structure of the Data_Unit for the request- and response-DLPDU ..... 82

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Figure 18 – Structuring of the protocol machines and adjacent layers in a DP-slave ..... 86
Figure 19 – Structuring of the protocol machines and adjacent layers in a DP-master (class 1) ..... 87
Figure 20 – Structuring of the protocol machines and adjacent layers in a DP-master (class 2) ..... 88
Figure 21 – Sequence of the communication between DP-master and DP-slave ..... 90
Figure 22 – Sequence of communication between DP-master (class 2) and DP-master (class 1) ..... 92
Figure 23 – Sequence of acyclic communication between DP-master (class 1) and DP-slave ..... 94
Figure 24 – Example for connection establishment on MS2 ..... 97
Figure 25 – Idle at master-side on MS2 ..... 98
Figure 26 – Idle at slave-side on MS2 ..... 99
Figure 27 – Example for connection establishment on MS2(server-side) ..... 239
Figure 28 – Structure of RM entries in the RM_Registry ..... 240

Table 1 – State machine description elements ..... 21
Table 2 – Description of state machine elements ..... 22
Table 3 – Conventions used in state machines ..... 22
Table 4 – APDU syntax ..... 24
Table 5 – Substitutions ..... 27
Table 6 – Alarm_Type range ..... 36
Table 7 – Status_Type value range ..... 36
Table 8 – Alarm_Specifier ..... 37
Table 9 – Range of Modul_Status_Entry (1-4) ..... 39
Table 10 – Error type ..... 41
Table 11 – Specification of the bits Lock_Req and Unlock_Req ..... 44
Table 12 – Range of Length_of_Manufacturer_Specific_Data if used in Chk_Cfg-REQ-PDU ..... 51
Table 13 – Range of Length_of_Manufacturer_Specific_Data if used in Get_Cfg-RES-PDU ..... 51
Table 14 – Values (codes) for data types ..... 53
Table 15 – Specification of the bits for Un-/Sync and Un-/Freeze ..... 54
Table 16 – Coding of the Function_Code/ Function_Num ..... 57
Table 17 – Coding of the Error_Code / Function_Num ..... 58
Table 18 – Values of Error_Decode ..... 58
Table 19 – Coding of Error_Code_1 at DPV1 ..... 59
Table 20 – Values of MDiag_Identifier ..... 60
Table 21 – Values for Area_Code_UpDownload ..... 62
Table 22 – Values for Area_CodeActBrct ..... 63
Table 23 – Values for Area_CodeAct ..... 63
Table 24 – Values for Data_rate ..... 64
Table 25 – Values for Slave_Type ..... 67
Table 26 – Values for Alarm_Mode ..... 68
Table 27 – Values for Subnet ..... 73
Table 28 – Values of reason code if instance is DLL ..... 73

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Table 29 – Values of reason code if instance is MS2 ..... 73
Table 30 – Values of Extended_Function_Num ..... 74
Table 31 – Values of FI_State ..... 76
Table 32 – Assignment of state machines ..... 85
Table 33 – Primitives issued by AP-Context to FSPMS ..... 100
Table 34 – Primitives issued by FSPMS to AP-Context ..... 102
Table 35 – FSPMS state table ..... 110
Table 36 – Functions used by the FSPMS ..... 135
Table 37 – Primitives issued by AP-Context to FSPMM1 ..... 137
Table 38 – Primitives issued by FSPMM1 to AP-Context ..... 139
Table 39 – FSPMM1 state table ..... 146
Table 40 – Functions used by the FSPMM1 ..... 172
Table 41 – Primitives issued by AP-Context to FSPMM2 ..... 173
Table 42 – Primitives issued by FSPMM2 to AP-Context ..... 175
Table 43 – FSPMM2 state table ..... 178
Table 44 – Functions used by the FSPMM2 ..... 189
Table 45 – Primitives issued by FSPMS to MSCY1S ..... 190
Table 46 – Primitives issued by MSCY1S to FSPMS ..... 191
Table 47 – Rules for DPV1_Status_1, DPV1_Status_2 and DPV1_Status_3 check ..... 193
Table 48 – MSCY1S state table ..... 198
Table 49 – Functions used by the MSCY1S ..... 219
Table 50 – Primitives issued by FSPMS to MSAC1S ..... 221
Table 51 – Primitives issued by MSAC1S to FSPMS ..... 222
Table 52 – Primitives issued by MSCY1S to MSAC1S ..... 222
Table 53 – Primitives issued by MSAC1S to MSCY1S ..... 222
Table 54 – Parameter used with primitives exchanged between MSAC1S and MSCY1S ..... 222
Table 55 – MSAC1S state table ..... 224
Table 56 – Functions used by the MSAC1S ..... 234
Table 57 – Primitives issued by FSPMS to SSCY1S ..... 234
Table 58 – Primitives issued by SSCY1S to FSPMS ..... 234
Table 59 – SSCY1S state table ..... 236
Table 60 – Functions used by the SSCY1S ..... 237
Table 61 – Primitives issued by FSPMS to MSRM2S ..... 238
Table 62 – Primitives issued by MSRM2S to FSPMS ..... 238
Table 63 – MSRM2S state table ..... 241
Table 64 – Primitives issued by FSPMS to MSAC2S ..... 244
Table 65 – Primitives issued by MSAC2S to FSPMS ..... 245
Table 66 – Primitives issued by MSRM2S to MSAC2S ..... 245
Table 67 – Primitives issued by MSAC2S to MSRM2S ..... 246
Table 68 – Parameter used with primitives exchanged with MSAC2S ..... 246
Table 69 – MSAC2S state table ..... 249
Table 70 – Primitives issued by MSCS1S to FSPMS ..... 260
Table 71 – MSCS1S state table ..... 262

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Table 72 – Primitives issued by FSPMM1 to MSCY1M ..... 263
Table 73 – Primitives issued by MSCY1M to FSPMM1 ..... 264
Table 74 – Parameters used with primitives exchanged between FSPMM1 and MSCY1M .... 264
Table 75 – MSCY1M state table ..... 267
Table 76 – Primitives issued by FSPMM1 to MSAL1M ..... 282
Table 77 – Primitives issued by MSAL1M to FSPMM1 ..... 282
Table 78 – Primitives issued by MSCY1M to MSAL1M ..... 282
Table 79 – Primitives issued by MSAL1M to MSCY1M ..... 283
Table 80 – Parameter used with primitives exchanged between MSAL1M and MSCY1M ..... 283
Table 81 – Possible values in the Alarm_State_Table ..... 284
Table 82 – MSAL1M state table ..... 287
Table 83 – Primitives issued by FSPMM1 to MSAC1M ..... 292
Table 84 – Primitives issued by MSAC1M to FSPMM1 ..... 292
Table 85 – Primitives issued by MSAL1M to MSAC1M ..... 293
Table 86 – Primitives issued by MSAC1M to MSAL1M ..... 293
Table 87 – Parameter used with primitives exchanged between MSAL1M and MSCY1M ..... 293
Table 88 – MSAC1M state table ..... 299
Table 89 – Primitives issued by FSPMM1 to MMAC1 ..... 305
Table 90 – Primitives issued by MMAC1 to FSPMM1 ..... 305
Table 91 – MMAC1 state table ..... 307
Table 92 – Primitives issued by FSPMM1 to MSCS1M ..... 312
Table 93 – Primitives issued by MSCS1M to FSPMM1 ..... 312
Table 94 – MSCS1M state table ..... 314
Table 95 – Primitives issued by FSPMM2 to MSAC2M ..... 316
Table 96 – Primitives issued by MSAC2M to FSPMM2 ..... 317
Table 97 – Parameters used with primitives exchanged with MSAC2M ..... 317
Table 98 – MSAC2M state table ..... 321
Table 99 – Primitives issued by FSPMM2 to MMAC2 ..... 331
Table 100 – Primitives issued by MMAC2 to FSPMM2 ..... 332
Table 101 – Parameters used with primitives exchanged with MMAC2 ..... 332
Table 102 – MMAC2 state table ..... 333
Table 103 – Primitives issued by FSPMS to DMPMS ..... 338
Table 104 – Primitives issued by DMPMS to FSPMS ..... 338
Table 105 – Primitives issued by MSCY1S to DMPMS ..... 339
Table 106 – Primitives issued by DMPMS to MSCY1S ..... 339
Table 107 – Primitives issued by DMPMS to SSCY1S ..... 340
Table 108 – Primitives issued by MSAC1S, MSRM2S, MSAC2S to DMPMS ..... 340
Table 109 – Primitives issued by DMPMS to MSAC1S, MSRM2S, MSAC2S ..... 340
Table 110 – Primitives issued by DMPMS to MSCS1S ..... 341
Table 111 – Primitives issued by DMPMS to DL ..... 341
Table 112 – Primitives issued by DL to DMPMS ..... 342
Table 113 – Parameters used with primitives exchanged with DMPMS ..... 343
Table 114 – DMPMS state table ..... 344

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Table 115 – Functions used by the DMPMS ..... 350
Table 116 – Primitives issued by FSPMM1 to DMPMM1 ..... 351
Table 117 – Primitives issued by DMPMM1 to FSPMM1 ..... 352
Table 118 – Primitives issued by MSCY1M to DMPMM1 ..... 352
Table 119 – Primitives issued by DMPMM1 to MSCY1M ..... 353
Table 120 – Primitives issued by MSAL1M, MSAC1M to DMPMM1 ..... 353
Table 121 – Primitives issued by DMPMM1 to MSAL1M, MSAC1M ..... 353
Table 122 – Primitives issued by MMAC1 to DMPMM1 ..... 354
Table 123 – Primitives issued by DMPMM1 to MMAC1 ..... 354
Table 124 – Primitives issued by MSCS1M to DMPMM1 ..... 354
Table 125 – Primitives issued by DMPMM1 to MSCS1M ..... 355
Table 126 – Primitives issued by DMPMM1 to DL ..... 355
Table 127 – Primitives issued by DL to DMPMM1 ..... 356
Table 128 – Parameters used with primitives exchanged with DMPMM1 ..... 357
Table 129 – Possible values of status ..... 358
Table 130 – DMPMM1 state table ..... 359
Table 131 – Functions used by the DMPMM1 ..... 367
Table 132 – Primitives issued by FSPMM2 to DMPMM2 ..... 368
Table 133 – Primitives issued by DMPMM2 to FSPMM2 ..... 368
Table 134 – Primitives issued by MSAC2M to DMPMM2 ..... 369
Table 135 – Primitives issued by DMPMM2 to MSAC2M ..... 369
Table 136 – Primitives issued by MMAC2 to DMPMM2 ..... 369
Table 137 – Primitives issued by DMPMM2 to MMAC2 ..... 369
Table 138 – Primitives issued by DMPMM2 to DL ..... 370
Table 139 – Primitives issued by DL to DMPMM2 ..... 370
Table 140 – Parameters used with primitives exchanged with DMPMM2 ..... 371
Table 141 – DMPMM2 state Table ..... 371
Table 142 – Functions used by DMPMM2 ..... 375
Table 143 – Bus parameter/reaction times for a DP-slave ..... 375

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

INTERNATIONAL ELECTROTECHNICAL COMMISSION

# INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

## Part 6-3: Application layer protocol specification – Type 3 elements

## FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.

NOTE 1 Use of some of the associated protocol types is restricted by their intellectual-property-right holders. In all cases, the commitment to limited release of intellectual-property-rights made by the holders of those rights permits a particular data-link layer protocol type to be used with physical layer and application layer protocols in Type combinations as specified explicitly in the profile parts. Use of the various protocol types in other combinations may require permission from their respective intellectual-property-right holders.

International Standard IEC 61158-6-3 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation.

This second edition cancels and replaces the first edition published in 2007. This edition constitutes a technical revision.

The main changes with respect to the previous edition are listed below:

- corrections, in Table 10 and Table 48;
- clarification in 6.9.1.2;
- expired patents deleted and new patents disclosed.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

This bilingual version published in 2012-07 corresponds to the English version published in 2010-08.

The text of this standard is based on the following documents:

| FDIS | Report on voting |
|---|---|
| 65C/607/FDIS | 65C/621/RVD |

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with ISO/IEC Directives, Part 2.

A list of all parts of the IEC 61158 series, published under the general title *Industrial communication networks – Fieldbus specifications*, can be found on the IEC web site.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be:

- reconfirmed;
- withdrawn;
- replaced by a revised edition, or
- amended.

NOTE 2 The revision of this standard will be synchronized with the other parts of the IEC 61158 series.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# INTRODUCTION

This part of IEC 61158 is one of a series produced to facilitate the interconnection of automation system components. It is related to other standards in the set as defined by the "three-layer" fieldbus reference model described in IEC/TR 61158-1.

The application protocol provides the application service by making use of the services available from the data-link or other immediately lower layer. The primary aim of this standard is to provide a set of rules for communication expressed in terms of the procedures to be carried out by peer application entities (AEs) at the time of communication. These rules for communication are intended to provide a sound basis for development in order to serve a variety of purposes:

- as a guide for implementors and designers;
- for use in the testing and procurement of equipment;
- as part of an agreement for the admittance of systems into the open systems environment;
- as a refinement to the understanding of time-critical communications within OSI.

This standard is concerned, in particular, with the communication and interworking of sensors, effectors and other automation devices. By using this standard together with other standards positioned within the OSI or fieldbus reference models, otherwise incompatible systems may work together in any combination.

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this document may involve the use of patents concerning Type 3 elements and possibly other types given in the normative elements of this standard.

The following patent rights for Type 3 have been announced by [SI]:

| Publication | Title |
|---|---|
| EP0604668-A1 (06.07.1994); EP0604668-B1 (18.02.1998) | Logical ring with monitoring of rotation time |
| EP0604669-A1 (06.07.1994); EP0604669-B1 (01.04.1998) | Bus system with monitoring of the activity state of participants |

IEC takes no position concerning the evidence, validity and scope of these patent rights.

The holder of these patent rights has assured the IEC that he/she is willing to negotiate licences either free of charge or under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of these patent rights is registered with IEC. Information may be obtained from:

[SI]: Siemens AG
CT IP L&T
Hr. Hans-Jörg Müller
Otto-Hahn-Ring 6
D-81739 Munich
Germany

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. IEC shall not be held responsible for identifying any or all such patent rights.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

ISO (www.iso.org/patents) and IEC (http://www.iec.ch/tctools/patent_decl.htm) maintain on-line data bases of patents relevant to their standards. Users are encouraged to consult the data bases for the most up to date information concerning patents.

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

# INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

## Part 6-3: Application layer protocol specification – Type 3 elements

## 1 Scope

### 1.1 General

The Fieldbus Application Layer (FAL) provides user programs with a means to access the fieldbus communication environment. In this respect, the FAL can be viewed as a "window between corresponding application programs."

This standard provides common elements for basic time-critical and non-time-critical messaging communications between application programs in an automation environment and material specific to Type 3 fieldbus. The term "time-critical" is used to represent the presence of a time-window, within which one or more specified actions are required to be completed with some defined level of certainty. Failure to complete specified actions within the time window risks failure of the applications requesting the actions, with attendant risk to equipment, plant and possibly human life.

This standard defines in an abstract way the externally visible behavior provided by the Type 3 fieldbus application layer in terms of

a) the abstract syntax defining the application layer protocol data units conveyed between communicating application entities,

b) the transfer syntax defining the application layer protocol data units conveyed between communicating application entities,

c) the application context state machine defining the application service behavior visible between communicating application entities; and

d) the application relationship state machines defining the communication behavior visible between communicating application entities; and.

The purpose of this standard is to define the protocol provided to

a) define the wire-representation of the service primitives specified in IEC 61158-5-3, and

b) define the externally visible behavior associated with their transfer.

This standard specifies the protocol of the Type 3 fieldbus application layer, in conformance with the OSI Basic Reference Model (ISO/IEC 7498-1) and the OSI Application Layer Structure (ISO/IEC 9545).

FAL services and protocols are provided by FAL application-entities (AE) contained within the application processes. The FAL AE is composed of a set of object-oriented Application Service Elements (ASEs) and a Layer Management Entity (LME) that manages the AE. The ASEs provide communication services that operate on a set of related application process object (APO) classes. One of the FAL ASEs is a management ASE that provides a common set of services for the management of the instances of FAL classes.

Although these services specify, from the perspective of applications, how request and responses are issued and delivered, they do not include a specification of what the requesting and responding applications are to do with them. That is, the behavioral aspects of the applications are not specified; only a definition of what requests and responses they can

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

send/receive is specified. This permits greater flexibility to the FAL users in standardizing such object behavior. In addition to these services, some supporting services are also defined in this standard to provide access to the FAL to control certain aspects of its operation.

### 1.2 Specifications

The principal objective of this standard is to specify the syntax and behavior of the application layer protocol that conveys the application layer services defined in IEC 61158-5-3.

A secondary objective is to provide migration paths from previously-existing industrial communications protocols. It is this latter objective which gives rise to the diversity of protocols standardized in parts of the IEC 61158-6 subparts.

### 1.3 Conformance

This standard does not specify individual implementations or products, nor does it constrain the implementations of application layer entities within industrial automation systems.

There is no conformance of equipment to the application layer service definition standard. Instead, conformance is achieved through implementation of this application layer protocol specification.

## 2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 60559, *Binary floating-point arithmetic for microprocessor systems*

IEC 61158-3-3:2010[1], *Industrial communication networks – Fieldbus specifications – Part 3-3: Data-link layer service definition – Type 3 elements*

IEC 61158-4-3:2010[1], *Industrial communication networks – Fieldbus specifications – Part 4-3: Data-link layer protocol specification – Type 3 elements*

IEC 61158-5-3:2010[1], *Industrial communication networks – Fieldbus specifications – Part 5-3: Application layer service definition – Type 3 elements*

IEC 61158-6-10:2010[1], *Industrial communication networks – Fieldbus specifications – Part 6-10: Application layer protocol specification – Type 10 elements*

ISO/IEC 10731, *Information technology – Open Systems Interconnection – Basic Reference Model – Conventions for the definition of OSI services*

ISO/IEC 7498-1, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*

ISO/IEC 8822, *Information technology – Open Systems Interconnection – Presentation service definition*

ISO/IEC 8824-1, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*

---

1 To be published

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

ISO/IEC 9545, *Information technology – Open Systems Interconnection – Application Layer structure*

## 3 Terms, definitions, abbreviations, symbols and conventions

### 3.1 Referenced terms and definitions

#### 3.1.1 ISO/IEC 7498-1 terms

For the purposes of this document, the following terms as defined in ISO/IEC 7498-1 apply:

a) application entity
b) application process
c) application protocol data unit
d) application service element
e) application entity invocation
f) application process invocation
g) application transaction
h) real open system
i) transfer syntax

#### 3.1.2 ISO/IEC 8822 terms

For the purposes of this document, the following terms as defined in ISO/IEC 8822 apply:

a) abstract syntax
b) presentation context

#### 3.1.3 ISO/IEC 9545 terms

For the purposes of this document, the following terms as defined in ISO/IEC 9545 apply:

a) application-association
b) application-context
c) application context name
d) application-entity-invocation
e) application-entity-type
f) application-process-invocation
g) application-process-type
h) application-service-element
i) application control service element

#### 3.1.4 ISO/IEC 8824-1 terms

For the purposes of this document, the following terms as defined in ISO/IEC 8824-1 apply:

a) object identifier
b) type

#### 3.1.5 Fieldbus Data Link Layer terms

For the purposes of this document, the following terms as defined in IEC 61158-3-3 and IEC 61158-4-3 apply.

a) DL-Time

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

b) DL-Scheduling-policy
c) DLCEP
d) DLC
e) DL-connection-oriented mode
f) DLPDU
g) DLSDU
h) DLSAP
i) link
j) network address
k) node address
l) node
m) scheduled

### 3.2 Additional definitions

**3.2.1**
**access protection**
limitation of the usage of an application object to one client

**3.2.2**
**address-assignment-table**
mapping of the client's internal I/O-Data object storage to the decentralised input and output data objects

**3.2.3**
**channel**
single physical or logical link of an input or output application object of a server to the process

**3.2.4**
**channel related diagnosis**
information concerning a specific element of an input or output application object, provided for maintenance purposes

EXAMPLE: validity of data

**3.2.5**
**configuration check**
comparison of the expected I/O-Data object structuring of the client with the real I/O-Data object structuring to the server in the start-up phase

**3.2.6**
**configuration fault**
an unacceptable difference between the expected I/O-Data object structuring and the real I/O-Data object structuring, as detected by the server

**3.2.7**
**configuration identifier**
representation of a portion of I/O Data of a single input- and/or output-module of a server.

**3.2.8**
**configuration identifier related diagnosis**
compromises all module specific data available at the server for maintenance purposes

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**3.2.9**
**control commands**
action invocations transferred from client to server to clear outputs, freeze inputs and/or synchronise outputs

**3.2.10**
**data consistency**
means for coherent transmission and access of the input- or output-data object between and within client and server

**3.2.11**
**Data-eXchange-Broadcast (DXB)**
service for conveyance of information between the DP-master (Class 1) and the assigned DP-slaves initiating the Publisher and Subscriber functionality in the DP-slaves

**3.2.12**
**default DL-address**
value 126 as an initial value for DL-address, which has to be changed (e.g. by assignment of an DL-address via the fieldbus) before operation with a DP-master (class 1)

**3.2.13**
**device related diagnosis**
compromises all data related to the whole DP-slave available at the server for maintenance purposes

**3.2.14**
**diagnosis information**
all data available at the server for maintenance purposes

**3.2.15**
**diagnosis information collection**
system diagnosis information that is assembled at the client side

**3.2.16**
**DP-master (class 1)**
a controlling device which controls several DP-slaves (field devices)

NOTE This is usually a programmable controller or a distributed control system.

**3.2.17**
**DP-master (class 2)**
controlling device which manages configuration data (parameter sets) and diagnosis data of a DP-master (Class 1), and that additionally performs all communication capabilities of a DP-master (Class 1)

**3.2.18**
**DP-slave**
field device that can be assigned to one DP-master (Class 1) as a provider for cyclic I/O data exchange, in addition acyclic functions and alarms could be provided

**3.2.19**
**DXB request**
cyclic data exchange service request with a specific DL service between the DP-master (Class 1) and the assigned DP-slaves initiating the Publisher functionality in the DP-slaves

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**3.2.20**
**DXB response**
cyclic data exchange service response not to the individual address of the requester (DP-master (Class 1)) but to the broadcast address (=127)

**3.2.21**
**freeze**
function at the DP-slaves for simultaneous data transfer between the input data object and the process

**3.2.22**
**group**
set of DP-slaves which perform a Freeze or Sync function

**3.2.23**
**I/O data**
object designated to be transferred cyclically for the purpose of processing

**3.2.24**
**ident number**
DP-master (Class 1) or DP-slave device type

**3.2.25**
**index**
address of an object within an application process

**3.2.26**
**isochronous mode**
special operation mode of a DP system that implies both a constant DP cycle with a fixed schedule of the cyclic and acyclic DP services, and the synchronisation of the application in the DP-master (Class 1) and the DP-slaves with this constant DP cycle

**3.2.27**
**master parameter set**
the configuration and parameterisation data of all DP-slaves that are assigned to the corresponding DP-master and the bus parameters

**3.2.28**
**module**
addressable unit inside the DP-slave

**3.2.29**
**process data**
object(s) which are already pre-processed and transferred acyclically for the purpose of information or further processing

**3.2.30**
**publisher**
role of a CR endpoint that transmits APDUs onto the Fieldbus for consumption by one or more subscribers

NOTE A publisher may not be aware of the identity or the number of subscribers and it may publish its APDUs using a dedicated CR.

**3.2.31**
**real configuration**
input and output data structure of a DP-slave, including definition of data consistency

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**3.2.32**
**subscriber**
role of a CREP in which it receives APDUs produced by a publisher

**3.2.33**
**slot**
address of a module within a DP-slave

**3.2.34**
**sync**
function at the DP-slaves for simultaneous data transfer between the output data object and the process

## 3.3 Abbreviations and symbols

| | |
|---|---|
| Ack | Acknowledgement |
| Add | Address |
| AP | Application Process |
| APDU | Application Protocol Data Unit |
| API | Application Process Identifier |
| AR | Application Relationship |
| AREP | Application Relationship End Point |
| ARL | Application Relationship List |
| ASE | Application Service Element |
| Cfg | Configuration |
| cnf | confirmation primitive |
| CREP | Communication Relationship End Point |
| CRL | Communication Relationship List |
| D_ | Destination |
| DLL | Data Link Layer |
| DLM | Data Link-management |
| DLSAP | Data Link Service Access Point |
| DLSDU | Data Link Service Data Unit |
| DMPM | Data Link Mapping Protocol Machine |
| DMPMM1 | Data Link Mapping Protocol Machine DP-master (Class 1) |
| DMPMM2 | Data Link Mapping Protocol Machine DP-master (Class 2) |
| DMPMS | Data Link Mapping Protocol Machine DP-slave |
| DSAP | Destination Service Access Point |
| DXB | Data eXchange Broadcast |
| FAL | Fieldbus Application Layer |
| FIFO | First In First Out |
| FSPMM1 | FAL Service Protocol Machine DP-master (Class 1) |
| FSPMM2 | FAL Service Protocol Machine DP-master (Class 2) |
| FSPMS | FAL Service Protocol Machine DP-slave |
| HSA | Highest Station Address |
| I&M | Identification and Maintenance Profile |
| ID | Identifier |
| IEC | International Electrotechnical Commission |
| ind | indication primitive |
| Inp | Input |
| ISO | International Organization For Standardization |
| IsoM | Isochronous Mode |
| L_sdu | Link Service Data Unit |
| lsb | least significant bit |
| msb | most significant bit |
| OSI | Open Systems Interconnection |
| Outp | Output |
| Param | Parameter |
| PDU | Protocol Data Unit |
| RD_ | Read |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| | |
|---|---|
| Rem | Remote |
| Req | Request (used to indicate an APDU type) |
| req | request primitive |
| REQ | Request to indicate a Request PDU |
| RES | Response to indicate a Response PDU |
| RM | Resource Manager |
| RPL_UPD | Reply Update |
| Rsp | Response (used to indicate an APDU type) |
| rsp | response primitive |
| S_ | Source |
| SAP | Service Access Point |
| SCL | Security Level |
| SD | Start Delimiter |
| SDN | Send Data with no Acknowledge |
| SDR | Station Delay Responder |
| Seq | Sequence |
| Src | Source |
| SRD | Send and Request Data with Reply |
| SSAP | Source Service Access Point |
| TR | Technical Report |
| USIF | User Interface |
| WD | Watchdog |

## 3.4 Conventions

### 3.4.1 General concept

The FAL is defined as a set of object-oriented ASEs. Each ASE is specified in a separate subclause. Each ASE specification is composed of three parts: its class definitions, its services, and its protocol specification. The first two are contained in IEC 61158-5-3. The protocol specification for each of the ASEs is defined in this standard.

The class definitions define the attributes of the classes supported by each ASE. The attributes are accessible from instances of the class using the Management ASE services specified in IEC 61158-5-3 standard. The service specification defines the services that are provided by the ASE.

This standard uses the descriptive conventions given in ISO/IEC 10731.

### 3.4.2 Abstract syntax conventions

PDUs are described as octets or groups of octets.

a) Groups of octets separated by a comma appear in the order they are transferred

If optional octets are not present the following octets appear without a gap

b) If octets or groups of octets are grouped within "{ }" the order is arbitrary

c) If octets or groups of octets are marked with "*" they may appear more than once. If it is used within a "{ }" section they may appear mixed with other octets or group of octets of this section.

d) Octets can be grouped or values can be assigned within "( )"

e) If octets or groups of octets are grouped within "[ ]" the group can be omitted

f) Complex APDUs may be built by means of substitutions (sub-structures)

g) Exclusive selections of octets or groups of octets are separated by "^"

NOTE 1 The formal PDU example

AP_PDU=Octet1,OctetGroup1,[Octet2],[Octet3],{[OctGroup2*],OctetGroup3^Octet4}

According to this the following variants are valid on the wire (non exhaustive):

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Variant 1: Octet1, OctetGroup1, Octet2, Octet3, OctetGroup2,OctetGroup3

Variant 2: Octet1, OctetGroup1, Octet2, Octet3, OctetGroup2, OctetGroup2, OctetGroup2,OctetGroup3

Variant 3: Octet1, OctetGroup1, OctetGroup2, OctetGroup2, OctetGroup2,OctetGroup3, OctetGroup2

Variant4: Octet1, OctetGroup1, OctetGroup2, OctetGroup3, OctetGroup2, OctetGroup2, OctetGroup2, OctetGroup2

Variant 5: Octet1, OctetGroup1, Octet3, Octet4

NOTE 2 The arbitrary order implies that groups of octets are characterised by a special header that is described within the coding rules.

NOTE 3 The APDU syntax implies that according to the maximum DLSDU a APDU must not exceed 244 octets in total.

### 3.4.3 Convention for the encoding of reserved bits and octets

The term "reserved" may be used to describe bits in octets or whole octets. All bits or octets that are reserved should be set to zero at the sending side and shall not be tested at the receiving side except it is explicitly stated or if the reserved bits or octets are checked by a state machine.

The term "reserved" may also be used to indicate that certain values within the range of a parameter are reserved for future extensions. In this case the reserved values should not be used at the sending side and shall not be tested at the receiving side except it is explicitly stated or if the reserved values are check by a state machine.

### 3.4.4 Conventions for the common coding s of specific field octets

APDUs may contain specific fields that carry information in a primitive and condensed way. These fields shall be coded in the order according to Figure 1.

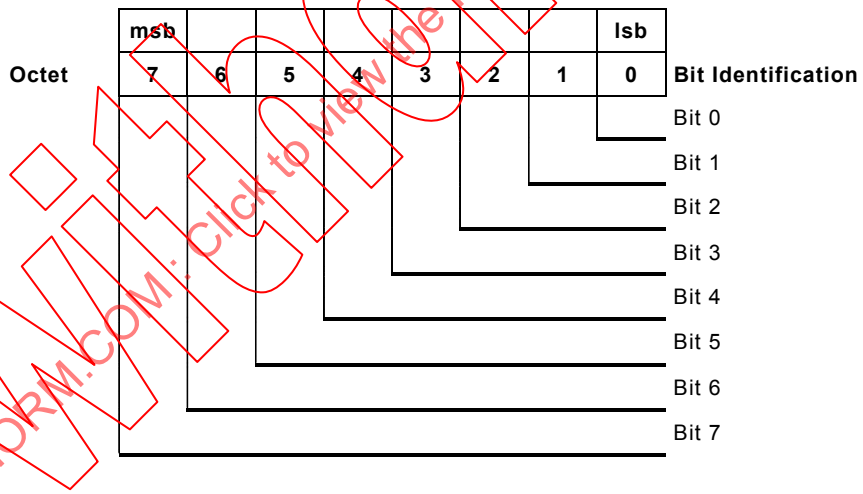



**Figure 1 – Common structure of specific fields**

- Bits may be grouped as group of bits. Each bit or group of bits shall be addressed by its Bit Identification (e.g. Bit 0, Bit 1 to 4). The position within the octet shall be according to the figure above. Alias names may be used for each bit or group of bits or they may be marked as reserved. The grouping of individual bits shall be in ascending order without gaps. The values for a group of bits may be represented as binary, decimal or hexadecimal values. This value shall only be valid for the grouped bits and can only represent the whole octet if all 8 bits are grouped. Decimal or hexadecimal values shall be transferred in binary values so that the bit with the highest number of the group represents the msb concerning the grouped bits.

EXAMPLE 1: Description and relation for the specific field octet

Bit 0: reserved.

Bit 1-3: Reason_Code The decimal value 2 for the Reason_Code means general error.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Bit 4-7: shall always set to one.

The octet that is constructed according to the description above looks as follows:

(msb) Bit 7 = 1,

Bit 6 = 1,

Bit 5 = 1,

Bit 4 = 1,

Bit 3 = 0,

Bit 2 = 1,

Bit 1 = 0,

(lsb) Bit 0 = 0.

The bit combination "0-1-0" for Bit 1-3 equals the decimal value 2.

## 3.5 Conventions used in state machines

### 3.5.1 State machine conventions

The protocol sequences are described by means of State Machines.

In state diagrams states are represented as boxes state transitions are shown as arrows. Names of states and transitions of the state diagram correspond to the names in the textual listing of the state transitions.

The textual listing of the state transitions is structured as follows, see also Table 1.

The first row contains the name of the transition.

The second row in define the current state.

The third row contains an optional event followed by Conditions starting with a “/” as first line character and finally followed by the Actions starting with a “=>” as first line character.

The last row contains the next state.

If the event occurs and the conditions are fulfilled the transition fires, i.e. the actions are executed and the next state is entered.

The layout of a Machine description is shown in Table 1. The meaning of the elements of a State Machine Description are shown in Table 2.

**Table 1 – State machine description elements**

| # | Current state | Event or condition => action | Next state |
|---|---|---|---|
| | | | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 2 – Description of state machine elements**

| Description element | Meaning |
|---|---|
| Current state<br>Next state | Name of the given states. |
| # | Name or number of the state transition. |
| Event | Name or description of the event. |
| /Condition | Boolean expression. The preceding "\" is not part of the condition. |
| => Action | List of assignments and service or function invocations. The preceding "=>" is not part of the action. |

The conventions used in the state machines are shown in Table 3.

**Table 3 – Conventions used in state machines**

| Convention | Meaning |
|---|---|
| := | Value of an item on the left is replaced by value of an item on the right. If an item on the right is a parameter, it comes from the primitive shown as an input event. |
| xxx | A parameter name.<br>Example:<br>Identifier := reason<br>means value of a 'reason' parameter is assigned to a parameter called 'Identifier.' |
| "xxx" | Indicates fixed value.<br>Example:<br>Identifier := "abc"<br>means value "abc" is assigned to a parameter named 'Identifier.' |
| = | A logical condition to indicate an item on the left is equal to an item on the right. |
| < | A logical condition to indicate an item on the left is less than the item on the right. |
| > | A logical condition to indicate an item on the left is greater than the item on the right. |
| <> | A logical condition to indicate an item on the left is not equal to an item on the right. |
| && | Logical "AND" |
| \|\| | Logical "OR" |
| for (Identifier := start_value to end_value)<br>actions<br>endfor | This construct allows the execution of a sequence of actions in a loop within one transition. The loop is executed for all values from start_value to end_value. |
| If (condition)<br>actions<br>else<br>actions | This construct allows the execution of alternative actions depending on some condition (which might be the value of some identifier or the outcome of a previous action) within one transition. The parts beginning with "else" can be ommitted if there is no action if the condition is not fulfilled. |

Readers are strongly recommended to refer to the subclauses for the AREP and CREP attribute definitions, the local functions, and the FAL-PDU definitions to understand protocol machines. It is assumed that readers have sufficient knowledge of these definitions and they are used without further explanations.

In addition the following description elements are used:

**Wildcard in names**

name_XXX: "XXX" is used as wildcard string for all names beginning with "name".

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

The typical use of a wildcard is an Event. In this context there are as many state transitions as possible events for this wildcard exists.

**Conditional macro**

<CONDITIONAL -MACRO-NAME>

<

CONDITION1: macrobody1

CONDITION2: macrobody2

…

>

CONDITION1, CONDITION2, etc. define all possible cases for the conditional macro. The "CONDITIONAL-MACRO-NAME" acts as place holder for the "macrocode" depending on the result of the condition.

**Replacement macro**

<REPLACEMENT-MACRO-NAME>

<

XXX= Name1 : macrobody1

XXX= Name2 : macrobody2

…

>

Name1, Name2, etc. define all possible cases for the replacement macro. The "REPLACEMENT-MACRO-NAME" acts as place holder for the "macrocode" depending on the value of the current use of the wildcard XXX.

Example:

<SERVICE_REQ_PARA>

<

XXX=Read: Para1, Para2

XXX=Write: Para1, Para2, Para3

>

when called in

MSAB_XXX.req(<SERVICE_REQ_PARA>)

will result in

MSAB_Read.req(Para1, Para2) or MSAB_Write.req(Para1, Para2, Para3)

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

# 4 FAL syntax description

## 4.1 APDU abstract syntax

Table 4 defines the abstract syntax of the DP Application Layer PDUs referred to as APDUs. The defined order of octets shall be used to convey the APDUs.

**Table 4 – APDU syntax**

| APDU name | APDU structure |
|---|---|
| DataExchange-REQ-PDU | [Outp_Data*] |
| DataExchange-RES-PDU | [Inp_Data*] |
| RD_Input-RES-PDU | [Inp_Data*] |
| RD_Output-RES-PDU | [Outp_Data*] |
| Chk_Cfg-REQ-PDU | {[Identifier_Format*], [Special_Identifier_Format*]^ [Extended_Special_Identifier_Format*]} |
| Get_Cfg-RES-PDU | {[Identifier_Format*], [Special_Identifier_Format*]^[Extended_Special_Identifier_Format*]} |
| Set_Prm-REQ-PDU | Station_status, WD_Fact_1, WD_Fact_2, min_TSDR, Ident_Number, Group_Ident, User_Prm_Data |
| Set_Ext_Prm-REQ-PDU | Structured_Prm_Data* |
| Diagnosis-RES-PDU | Station_status_1, Station_status_2, Station_status_3, Diag_Master_Add, Ident_Number, {[Device_Related_Diagnosis*]^([Alarm], [Status*],[Modul_Status],[DXB_Link_Status], { [Red_Status] ^[PrmCmdAck]}), [Identifier_Related_Diagnosis], [Channel_Related_Diagnosis*], [Revision_Number]}<br>The field Device_Related_Diagnosis may be present if DPV1=FALSE, otherwise if DPV1=TRUE it shall not be present. In this case the fields Alarm, Status, Modul_Status may be present. |
| Set_Slave_Add-REQ-PDU | New_Slave_Add, Ident_Number, No_Add_Chg, [Rem_Slave_Data*] |
| Global_Control-REQ-PDU | Control_Command, Group_Select |
| Clock-Value-PDU | Clock_value_time_event, Clock_value_previous_TE, Clock_value_status1, Clock_value_status2 |
| Initiate-REQ-PDU | Function_Num(0x57), reserved (3 Octets), Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Para |
| Initiate-RES-PDU | Function_Num(0x57), Max_Len_Data_Unit, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param |
| Initiate-NRS-PDU | Function_Num(0xD7), Error_Decode, Error_Code_1, Error_Code_2 |
| Abort-REQ-PDU | Function_Num(0x58), Subnet, Instance_Reason_Code |
| Read-REQ-PDU | Function_Num(0x5E), Slot_Number, Index (0 .. 254), Length |
| Read-RES-PDU | Function_Num(0x5E), Slot_Number, Index (0 .. 254), Length, [Data*] |
| Read-NRS-PDU<br>Pull-NRS-PDU | Function_Num(0xDE), Error_Decode, Error_Code_1, Error_Code_2 |
| Write-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (0 .. 254), Length, [Data*] |
| Write-RES-PDU | Function_Num(0x5F), Slot_Number, Index (0 .. 254), Length |
| Write-NRS-PDU<br>Initiate_Load_NRS-PDU<br>Push-NRS-PDU<br>Terminate_Load-NRS-PDU<br>Start-NRS-PDU<br>Stop-NRS-PDU<br>Resume-NRS-PDU<br>Reset-NRS-PDU<br>Call-NRS-PDU<br>Get_FI_State-NRS-PDU | Function_Num(0xDF), Error_Decode, Error_Code_1, Error_Code_2 |
| Alarm_Ack-REQ-PDU | Function_Num(0x5C), Slot_Number, Alarm_Type, Alarm_Specifier |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| APDU name | APDU structure |
|---|---|
| Alarm_Ack-RES-PDU | Function_Num(0x5C), Slot_Number, Alarm_Type, Alarm_Specifier |
| Alarm_Ack-NRS-PDU | Function_Num(0xDC), Error_Decode, Error_Code_1, Error_Code_2 |
| Idle-REQ-PDU | Function_Num(0x48) |
| Idle-RES-PDU | Function_Num(0x48) |
| Data_Transport-REQ-PDU | Function_Num(0x51), Slot_Number, Index, Length, [Data*] |
| Data_Transport-RES-PDU | Function_Num(0x51), Slot_Number, Index, Length, [Data*] |
| Data_Transport-NRS-PDU | Function_Num(0xD1), Error_Decode, Error_Code_1, Error_Code_2 |
| Initiate_Load-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x00), Options, LR_Index, LR_Length, Intersegment_Request_Timeout, [User_Specific] |
| Initiate_Load-RES-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x00), Max_Segment_Length, LR_Index, LR_Length, Max_Response_Delay |
| Push-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x01), Options, Sequence_Number, LR_Data |
| Pull-REQ-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length |
| Pull-RES-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x02), Options, Sequence_Number, LR_Data |
| Terminate_Load-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x03), reserved (1 octet), reserved (2 Octets) |
| Start-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x04), reserved (1 octet), FI_Index, [Execution_Argument] |
| Stop-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x05), reserved (1 octet), FI_Index |
| Resume-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x06), reserved (1 octet), FI_Index, [Execution_Argument] |
| Reset-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x07), reserved (1 octet), FI_Index |
| Call-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x08), Entity Number, FI_Index (=0…64999), [Execution_Argument]<br>Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x08), Entity Number, FI_Index (=65000..65199), [I MData_Exexcution_Argument] |
| Call-RES-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x08), Entity Number, FI_Index (=0…64999), [Result_Argument]<br>Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x08), Entity Number, FI_Index (=65000…65199), [IMData_Result_Argument] |
| Get_FI_State-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x09), reserved (1 octet), FI_Index |
| Get_FI_State-RES-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x09), reserved (1 octet), FI_Index, FI_State |
| Push-RES-PDU<br>Terminate_Load-RES-PDU<br>Start-RES-PDU<br>Stop-RES-PDU<br>Resume-RES-PDU<br>Reset-RES-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length |
| RM-REQ-PDU | Function_Num(0x56), Server_SAP, Send_Timeout |
| Get_Master_Diag-REQ-PDU | Function_Num(0x41), MDiag_Identifier |
| Get_Master_Diag-RES-PDU | Function_Num(0x41), MDiag_Identifier(=0..125), Diagnosis-RES-PDU<br>Function_Num(0x41), MDiag_Identifier(=126), System_Diagnosis,<br>Function_Num(0x41), MDiag_Identifier(=127), Master_Status<br>Function_Num(0x41), MDiag_Identifier(=128), Data_Transfer_List |
| Start_Seq-REQ-PDU | Function_Num(0x42), Area_CodeUpDownload, Timeout |
| Start_Seq-RES-PDU | Function_Num(0x42), Area_CodeUpDownload, Timeout, Max_Len_Data_Unit |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| APDU name | APDU structure |
|---|---|
| Download-REQ-PDU | Function_Num(0x43), Area_CodeUpDownload(=0..125), Add_Offset, DP_Slave_Parameter_Set<br>Function_Num(0x43), Area_CodeUpDownload(=127), Add_Offset, Bus_Parameter_Set<br>Function_Num(0x43), Area_CodeUpDownload(=129), Add_Offset, Statistic_Counters<br>Function_Num(0x43), Area_CodeUpDownload(=136 to 139), Add_Offset, Master_Parameter_Set |
| Download-RES-PDU | Function_Num(0x43), Area_CodeUpDownload, Add_Offset |
| Upload-REQ-PDU | Function_Num(0x44), Area_CodeUpDownload, Add_Offset, Data_Len |
| Upload-RES-PDU | Function_Num(0x44), Area_CodeUpDownload(=0..125), Add_Offset, DP_Slave_Parameter_Set<br>Function_Num(0x44), Area_CodeUpDownload(=127), Add_Offset, Bus_Parameter_Set<br>Function_Num(0x44), Area_CodeUpDownload(=129), Add_Offset, Statistic_Counters<br>Function_Num(0x43), Area_CodeUpDownload(=136 to 139), Add_Offset, Master_Parameter_Set |
| End_Seq-REQ-PDU | Function_Num(0x45) |
| End_Seq-RES-PDU | Function_Num(0x45) |
| Act_Para_Brct-REQ-PDU | Function_Num(0x46), Area_CodeActBrct |
| Act_Param-REQ-PDU | Function_Num(0x47), Area_CodeAct, Activate |
| Act_Param-RES-PDU | Function_Num(0x47), Area_CodeAct, Activate |
| ZERO-PDU | Octet1(0) |
| NULL-PDU | |

NOTE 1 If a APDU consists only of user data octets the distinction is made by use of different Data Link Layer service access points (see Protocol Machines).

NOTE 2 A ZERO-PDU contains one octet with all bits set to zero. This APDU is used to indicate diagnosis from a device without inputs.

NOTE 3 A NULL-PDU contains no octets. It is used to submit no DLSDU to data link layer.

Table 5 defines structures for substitutions of elements of the APDU structures shown in Table 4.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 5 – Substitutions**

| Substitution name | Structure |
|---|---|
| Identifier_Format | Cfg_Identifier |
| Special_Identifier_Format | Special_Cfg_Identifier, [Length_Octet] , [Length_Octet], [Manufacturer_Specific_Data*]<br>Length_Octet fields shall always start with fields that indicate output data if present. |
| Extended_Special_Identifier_Format | Special_Cfg_Identifier,[ Extended_Length_Octet], [Extended_Length_Octet], [Data_Type*], [Manufacturer_Specific_Data*]<br>Extended_Length_Octet fields shall always start with fields that indicate output data if present.<br>The field Data_Type shall only be omitted in case of an empty slot. Otherwise it shall be present in relation to the length fields. |
| Device_Related_Diagnosis | Header_Octet, Diagnosis_User_Data* |
| Identifier_Related_Diagnosis | Header_Octet, Identifier_Diagnosis_Data_Array |
| Channel_Related_Diagnosis | Identifier_Number, (Channel_Number, Type_of_Diagnosis)* |
| Alarm | Header_Octet, Alarm_Type, Slot_Number, Alarm_Specifier, [Diagnosis_User_Data*] |
| Status | Header_Octet, Status_Type, Slot_Number, Status_Specifier, Diagnosis_User_Data*] |
| Modul_Status | Header_Octet, Status_Type(=Modul_Status), Slot_Number(=0), Status_Specifier, Modul_Status_Array |
| DXB_Link_Status | Header_Octet, Status_Type(=DXB_Link_Status), Slot_Number(=0), Status_Specifier, (Publisher_Address, Publisher_Status)* |
| PrmCmdAck | Header_Octet, Status_Type(=PrmCmdAck), Slot_Number(=0), Status_Specifier, Function, Red_Status1, Red_Status2, Red_Status3 |
| Red_Status | Header_Octet, Status_Type(=Red_Status), Slot_Number(=0), Status_Specifier, Function, Red_Status1, Red_Status2, Red_Status3 |
| User_Prm_Data | [DPV1_Status_1, DPV1_Status_2, DPV1_Status_3], [User_Prm_Data_Element*]^[Structured_Prm_Data*]<br>The fields DPV1_Status_1, DPV1_Status_2 and DPV1_Status_3 shall be present if DPV1=TRUE.<br>Structured_Prm_Data shall be present if the DPV1_Status_3.Prm_Structure is TRUE, otherwise it shall not be used. |
| Structured_Prm_Data | Structure_Length, Structure_Type, Slot_Number, reserved (1 octet), User_Prm_Data*<br>Special Case DXB LinkTable:<br>Structure_Length, Structure_Type(=3), Slot_Number(=0), reserved (1 octet), Version(=1), (Publisher_Addr, Publisher_Length, Sample_Offset, Sample_Length)*<br>Special Case DXB-Subscribertable:<br>Structure_Length, Structure_Type(=7), Slot_Number(=0), reserved (1 octet), Version(=1), (Publisher_Addr, Publisher_Length, Sample_Offset, Dest_Slot_Number, Offset_Data_Area, Sample_Length)*<br>Special Case IsoM Parameter<br>Structure_Length, Structure_Type(=4), Slot_Number(=0), reserved (1 octet), Version(=1), TBASE_DP, TDP, TMAPC, TBASE_IO, TI, TO, TDX, TPLL_W, TPLL_D<br>Special Case PrmCmd<br>Structure_Length, Structure_Type(=2), Slot_Number(=0), Specifier, Function, Properties, Output_Hold_Time<br>Special Case Time AR Parameter:<br>Structure_Length, Structure_Type(=8), Slot_Number(=0), reserved (1 octet), Clock Sync Interval, [CS Delay Time]<br>Special Case F-Parameter<br>Structure_Length, Structure_Type(=5), Slot_Number, reserved (1 octet), User_Prm_Data* |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| Substitution name | Structure |
|---|---|
| Features_Supported | Features_Supported_1, Features_Supported_2 |
| Profile_Features_Supported | Profile_Features_Supported_1, Profile_Features_Supported_1 |
| Add_Addr_Para | S_Type, S_Len, D_Type, D_Len, S_API, S_SCL, [S_Network_Address], [S_MAC_Address], D_API, D_SCL, [D_Network_Address], [D_MAC_Address] |
| Master_Status | USIF_State, Ident_Number, Hardware_Release_DP, Firmware_Release_DP, Hardware_Release_User, Firmware_Release_User, reserved (9 Octets) |
| DP_Slave_Parameter_Set | Slave_Para_Len, SI_Flag, Slave_Type, Max_Diag_Data_Len, Max_Alarm_Len, Max_Channel_Data_Length, Diag_Upd_Delay, Alarm_Mode, Add_SI_Flag, MS1_Timeout, reserved (4 Octets), Prm_Data_Len, Prm_Data, Cfg_Data_Len, Cfg_Data, Add_Tab_Len, Add_Tab, Slave_User_Data_Len, Slave_User_Data, Ext_Prm_Data_Len, [Ext_Prm_Data] |
| Bus_Parameter_Set | Bus_Para_Len, DL_Add, Data_rate, $T_{SL}$, min $T_{SDR}$, max $T_{SDR}$, $T_{QUI}$, $T_{SET}$, $T_{TR}$, G, HSA, max_retry_limit, Bp_Flag, Min_Slave_Interval, Poll_Timeout, Data_Control_Time, Alarm_Max, Max_User_Global_Control, reserved (4 Octets), Master_User_Data_Len, Master_Class2_Name, Master_User_Data*, $T_{CT}$, max$T_{SH}$ |
| Master_Parameter_Set | Bus_Parameter_Set, DP_Slave_Parameter_Set* |
| Statistic_Counters | Counter_Group* |
| Counter_Group | if Add_Offset 0..126: DLPDU_sent_count(Add_Offset), Error_count(Add_Offset)<br>if Add_Offset 127: SD_sent_count, SD_error_count |
| Add_Tab | [Number_of_Entries], [Add_Tab_Entry_Header, I/O_Data_Length, IO_Config_Address, Host_Address]* |
| IMData_Execution_Argument | [I&M1] ^ [I&M2] ^ [I&M3] ^ [I&M4] ^ [I&M_Profile_Specific_x] ^ [I&M_Manufacturer_Specific_x] |
| IMData_Result_Argument | I&M0 ^ [I&M1] ^ [I&M2] ^ [I&M3] ^ [I&M4] ^ [I&M_Profile_Specific_x] ^ [I&M_Manufacturer_Specific_x] |
| I&M0 | IM_Header, IM_Manufacturer_ID, OrderID, IM_Serial_Number, IM_Hardware_Revision, IM_Software_Revision, IM_Revision_Counter, IM_Profile_ID, IM_Profile_Specific_Type, IM_Version, IM_Supported |
| I&M1 | IM_Header, IM_Tag_Function, IM_Tag_Location |
| I&M2 | IM_Header, IM_Date |
| I&M3 | IM_Header, IM_Descriptor |
| I&M4 | IM_Header, IM_Signature |
| I&M_Profile_Specific_x | IM_Header, IM_Profile_Specific_Data |
| IM_Version | IM_Version_Major, IM_Version_Minor |
| IM_Software_Revision | SWRevisionPrefix,IM_SWRevision_Functional_Enhancement, IM_SWRevision_Bug_Fix, IM_SWRevision_Internal_Change |

NOTE The DP_Slave_Parameter_Set and the Bus_Parameter_Set may exceed the maximum APDU size to a size up to 64 Koctets. The conveyance of the data is therefore segmented by means of a window addressed with the parameter Add_Offset.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

## 4.2 Data types

### 4.2.1 Notation for the Boolean type

Boolean ::= BOOLEAN -- TRUE if the value is non-zero.

-- FALSE if the value is zero.

#### 4.2.2 Notation for the Integer type

Integer8::= INTEGER (-128..+127) -- range $-2^7 <= I <= 2^7-1$

Integer16 ::= INTEGER (-32768..+32767) -- range $-2^{15} <= I <= 2^{15}-1$

Integer32 ::= INTEGER -- range $-2^{31} <= I <= 2^{31}-1$

Integer64 ::= INTEGER -- range $-2^{63} <= I <= 2^{63}-1$

#### 4.2.3 Notation for the Unsigned type

Unsigned8 ::= INTEGER (0..255) -- range $0 <= I <= 2^8-1$

Unsigned16 ::= INTEGER (0..65535) -- range $0 <= I <= 2^{16}-1$

Unsigned32 ::= INTEGER -- range $0 <= I <= 2^{32}-1$

Unsigned64 ::= INTEGER -- range $0 <= I <= 2^{64}-1$

#### 4.2.4 Notation for the Floating Point type

Floating32 ::= BIT STRING SIZE (4) -- IEC 60559 Single precision

Floating64 ::= BIT STRING SIZE (8) -- IEC 60559 Double precision

#### 4.2.5 Notation for the OctetString type

OctetString ::= OCTET STRING -- For generic use

#### 4.2.6 Notation for VisibleString type

VisibleString2 ::= VISIBLE STRING -- For generic use

#### 4.2.7 Notation for BinaryDate type

BinaryDate ::= OctetString7

#### 4.2.8 Notation for TimeOfDay type

TimeOfDay with date indication ::= OctetString6

TimeOfDay without date indication ::= OctetString4

#### 4.2.9 Notation for TimeDifference type

TimeDifference with date indication::= OctetString6

TimeDifference without date indication::= OctetString4

#### 4.2.10 Notation for Network Time type

Network Time ::= OctetString8

#### 4.2.11 Notation for Network Time Difference type

Network Time Difference::= OctetString8

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# 5 Transfer syntax

## 5.1 Coding of basic data types

### 5.1.1 Encoding of a Boolean value

a) The encoding of a Boolean value shall be primitive. The ContentsOctets shall consist of a single octet.

b) If the Boolean value is FALSE, the ContentsOctets shall be 0 (zero). If the Boolean value is TRUE, the ContentsOctets shall be 0xff.

### 5.1.2 Encoding of an Integer value

a) The encoding of a fixed-length Integer value of Integer8, Integer16, and Integer32 types shall be primitive, and the ContentsOctets shall consist of exactly one, two, or four octets, respectively.

b) The ContentsOctets shall be a two's complement binary number equal to the integer value, and consist of bits 7 to 0 of the first octet, followed by bits 7 to 0 of the second octet, followed by bits 7 to 0 of each octet in turn up to and including the last octet of the ContentsOctets.

NOTE The value of a two's complement binary number is derived by numbering the bits in the ContentsOctets, starting with bit 0 of the last octet as bit zero and ending the numbering with bit 7 of the first octet. Each bit is assigned a numerical value of $2^N$, where N is its position in the above numbering sequence. The value of the two's complement binary number is obtained by adding the numerical values assigned to each bit for those bits which are set to one, excluding bit 7 of the first octet, and then reducing this value by the numerical value assigned to bit 7 of the first octet if that bit is set to one.

### 5.1.3 Encoding of an Unsigned value

- The encoding of a fixed-length Unsigned value of Unsigned8, Unsigned16, and Unsigned32 types shall be primitive, and the ContentsOctets shall consist of exactly one, two, or four octets, respectively.
- The ContentsOctets shall be a binary number equal to the Unsigned value, and consist of bits 7 to 0 of the first octet, followed by bits 7 to 0 of the second octet, followed by bits 7 to 0 of each octet in turn up to and including the last octet of the ContentsOctets.

NOTE The value of a binary number is derived by numbering the bits in the ContentsOctets, starting with bit 0 of the last octet as bit zero and ending the numbering with bit 7 of the first octet. Each bit is assigned a numerical value of $2^N$, where N is its position in the above numbering sequence. The value of the binary number is obtained by adding the numerical values assigned to each bit for those bits which are set to one.

### 5.1.4 Encoding of a Floating-Point value

a) The encoding of a Floating32 value shall be primitive, and the ContentsOctets shall consist of exactly four octets.

b) The ContentsOctets shall contain floating-point values defined in conformance with IEC 60559. The sign is encoded in bit 7 of the first octet. It is followed by the exponent starting from bit 6 of the first octet, and then the mantissa starting from bit 6 of the second octet for Floating32.

### 5.1.5 Encoding of a Visible String value

a) The encoding of a variable length VisibleString value shall be primitive.

b) There is no Length field; the length is encoded implicitly.

c) The ContentsOctets shall be a sequence of octets. The leftmost string element is encoded in the first octet, followed by the second octet, followed by each octet in turn up to and including the last octet as rightmost of the ContentsOctets.

### 5.1.6 Encoding of an Octet String value

a) The encoding of a variable length OctetString value shall be primitive.

b) There is no Length field; the length is encoded implicitly.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

c) The ContentsOctets shall be a sequence of octets. The leftmost string element is encoded in the first octet, followed by second octet, followed by each octet in turn up to and including the last octet as rightmost of the ContentsOctets.

### 5.1.7 Encoding of a BinaryDate value

a) The encoding of a BinaryDate value shall be primitive.

b) The ContentsOctets shall be equal in value to the octets in the data value, as shown in Figure 2:

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | 0…59 999 ms |
| 2 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| 3 | RSV | RSV | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 0…59 min |
| 4 | SU | RSV | RSV | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 0…23 hours |
| | day of week | | | day of month | | | | | 1…7 day of week |
| 5 | $2^{2}$ | $2^{1}$ | $2^{0}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 1…31 day of month |
| 6 | RSV | RSV | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 1…12 months |
| 7 | RSV | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 0 … 50 years 2000 to 2050<br>51 … 99 years 1951 to 1999 |
| | msb | | | | | | | | |

NOTE To avoid the Y2K problem the values of the year are from 0 to 50 for the years 2000 to 2050 and from 51 to 99 for the years 1951 to 1999. This is according to the British Standards Institution (BSi) DISC PD2000-1:1998 “A Definition of Year 2000 Conformance Requirements” Rule 3 (b).

**Figure 2 – Coding of the data type BinaryDate**

### 5.1.8 Encoding of a TimeOfDay with and without date indication value

a) The encoding of a TimeOfDay with and without date indication value shall be primitive.

b) The ContentsOctets shall be equal in value to the octets in the data value, as shown in Figure 3:

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | 0 | 0 | 0 | 0 | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | number of milliseconds since midnight |
| 2 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | |
| 3 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | |
| 4 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| 5 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | number of days since 01.01.84 only with date indication |
| 6 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| | msb | | | | | | | | |

**Figure 3 – Encoding of TimeOfDay value**

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.1.9 Encoding of a Time Difference with and without date indication value

a) The encoding of a Time Difference with and without date indication value shall be primitive.

b) The ContentsOctets shall be equal in value to the octets in the data value, as shown in Figure 4:

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | |
| 2 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | milliseconds |
| 3 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | |
| 4 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| 5 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | days |
| 6 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | only with date indication |
| | msb | | | | | | | | |

**Figure 4 – Encoding of Time Difference value**

### 5.1.10 Encoding of a Network Time value

a) The encoding of a Network Time value shall be primitive.

b) The ContentsOctets shall be equal in value to the octets in the data value, as shown in Figure 5:

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | Sec. since 1.1.1900 0.00,00 |
| 2 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | or since 7.2. 2036 6.28,16 |
| 3 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | when time value less than |
| 4 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 0x9dff4400.00000000 |
| 5 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | Fraction Part |
| 6 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | of Seconds |
| 7 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | one unit is |
| 8 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 1/(2^32) s |
| | msb | | | | | | | | |

**Figure 5 – Encoding of Network Time value**

### 5.1.11 Encoding of a Network Time Difference value

a) The encoding of a Network Time Difference value shall be primitive.

b) The ContentsOctets shall be equal in value to the octets in the data value, as shown in Figure 6.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | |
| 2 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | Seconds as |
| 3 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | Integer32 |
| 4 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| 5 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | Fraction Part |
| 6 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | of Seconds |
| 7 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | one unit is |
| 8 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 1/(2^32) s |
| | msb | | | | | | | | |

**Figure 6 – Encoding of Network Time Difference value**

### 5.1.12 Encoding of a Null value

a) The encoding of a NULL value shall be primitive.

b) The ContentsOctet shall be omitted.

## 5.2 Coding section related to data exchange PDUs

### 5.2.1 Coding of the field Outp_Data

One of the following data types shall be used for each Outp_Data field:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.2.2 Coding of the field Inp_Data

One of the following data types shall be used for each Inp_Data field:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

NOTE Data types with variable length (Visible String, Octet String, TimeOfDay, Time Difference) for Inp_Data or Outp_Data are only once present in the DataExchange-RES-PDU or DataExchange-REQ-PDU. The user data correspond to the contents of the Configuration-PDU.

## 5.3 Coding section related to slave diagnosis PDUs

### 5.3.1 Coding of the field Station_status_1

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: Diag.Station_Non_Existent**
Shall be set to zero in case of Diagnoses-RES-PDU

Value (0): means FALSE if Get_Master_Diag-RES-PDU

Value (1): means TRUE if Get_Master_Diag-RES-PDU

**Bit 1: Diag.Station_Not_Ready**
FALSE (0), TRUE (1)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Bit 2: Diag.Cfg_Fault (Configuration Fault)**
FALSE (0), TRUE (1)

**Bit 3: Diag.Ext_Diag (Extended Diagnosis)**
FALSE (0), TRUE (1)

**Bit 4: Diag.Not_Supported**
FALSE (0), TRUE (1)

**Bit 5: Diag.Invalid_Slave_Response**
Shall be set to zero if Diagnoses-RES-PDU

Value (0): means FALSE if Get_Master_Diag-RES-PDU

Value (1): means TRUE if Get_Master_Diag-RES-PDU

**Bit 6: Diag.Prm_Fault (Parameter Fault)**
FALSE (0), TRUE (1)

**Bit 7: Diag.Master_Lock**
Shall be set to zero if Diagnoses-RES-PDU

Value (0): means FALSE if Get_Master_Diag-RES-PDU

Value (1): means TRUE if Get_Master_Diag-RES-PDU

### 5.3.2 Coding of the field Station_status_2

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: Diag.Prm_Req (Parameterisation requested)**
FALSE (0), TRUE (1)

**Bit 1: Diag.Stat_Diag (Static Diagnosis)**
FALSE (0), TRUE (1)

**Bit 2: DP (DP Protocol)**
Shall always set to ones

**Bit 3: Diag.WD_On (Watchdog on)**
FALSE (0), TRUE (1)

**Bit 4: Diag.Freeze_Mode**
FALSE (0), TRUE (1)

**Bit 5: Diag.Sync_Mode (Synchronisation Mode)**
FALSE (0), TRUE (1)

**Bit 6: reserved**

**Bit 7: Diag.Deactivated**
Shall be set to zero if Diagnoses-RES-PDU

Value (0): means FALSE if Get_Master_Diag-RES-PDU

Value (1): means TRUE if Get_Master_Diag-RES-PDU

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.3.3 Coding of the field Station_status_3

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 6: reserved**

**Bit 7: Diag.Ext_Diag_Overflow (Overflow of extended Diagnosis)**
FALSE (0), TRUE (1)

### 5.3.4 Coding of the field Diag_Master_Add

This field shall be coded as data type Unsigned8 with the following values:

**Decimal (0-125)**
means the address of the DP-master (Class 1) which has parameterised this DP-slave

**Decimal (255)**
means DP-slave has not been parameterised or has got invalid parameterisation.

**Decimal (126-254)**
not allowed

### 5.3.5 Coding of the field Ident_Number

This field shall be coded as data type Unsigned16.

### 5.3.6 Coding of the field Header_Octet

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 5: Block_Length**
It shall contain the length of the diagnosis block. The Header_Octet itself shall always be counted.

For the base diagnosis format of the Device Related Diagnosis the values shall be in the range of decimal 2 to 63. For the base diagnosis format of the Identifier Related Diagnosis the values shall be in the range of decimal 2 to 32. For the DPV1 diagnosis format for the Device Related Diagnosis the values shall be in the range of decimal 4 to 63.

NOTE The Identifier Related Diagnoses is limited to 32 because the identifier itself are limited to 244.

**Bit 6 to 7: Selection**
Decimal (0): means Device Related Diagnosis

Decimal (1): means Identifier Related Diagnosis

### 5.3.7 Coding of the field Alarm_Type

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 6: Alarm_Type**
The Alarm_Type shall contain the values according to Table 6.

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 6 – Alarm_Type range**

| Value (decimal) | Meaning |
|---|---|
| 0 | reserved |
| 1 | Diagnostic_Alarm |
| 2 | Process_Alarm |
| 3 | Pull_Alarm |
| 4 | Plug_Alarm |
| 5 | Status_Alarm |
| 6 | Update_Alarm |
| 7 – 31 | reserved |
| 32 – 126 | manufacturer-specific |
| 127 | reserved |

**Bit 7: shall be zero (identifying Alarm)**

### 5.3.8 Coding of the field Status_Type

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 6: Status_Type**
The Status_Type shall contain values according to Table 7.

**Table 7 – Status_Type value range**

| Value (decimal) | Meaning |
|---|---|
| 0 | reserved |
| 1 | Status_Message |
| 2 | Modul_Status |
| 3 | DXB_Link_Status |
| 4 to 29 | reserved |
| 30 | PrmCmdAck |
| 31 | Red_Status |
| 32 to 126 | Manufacturer-specific |
| 127 | reserved |

**Bit 7: shall be one (identifying Status)**

### 5.3.9 Coding of the field Slot_Number

This field shall be coded as data type Unsigned8. The value shall be taken from the range zero to 254. The value 255 is reserved.

### 5.3.10 Coding of the field Alarm_Specifier

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Bit 0 to 1: Alarm_Specifier**
This bit group shall contain the alarm specifier with the values according to Table 8.

**Table 8 – Alarm_Specifier**

| Value (decimal) | Meaning | Comment |
|---|---|---|
| 0 | No further differentiation | |
| 1 | Error appears and Slot disturbed | the slot generates an alarm due to an error |
| 2 | Error disappears and Slot is okay | the slot generates an alarm and indicates that the slot has no further errors |
| 3 | Error disappears and Slot is still disturbed | the slot generates an alarm and indicates that the slot has still further errors |

**Bit 2: Additional_Acknowledge**
Value (0): means no additional acknowledge used

Value (1): means additional acknowledge is used

**Bit 3 to 7: Sequence_Number**
The value range shall be from zero to 31 (decimal).

### 5.3.11 Coding of the field Status_Specifier

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 1: Status_Specifier**
Decimal (0): means no further differentiation

Decimal (1): means status appears

Decimal (2): means status disappears

Decimal (3): reserved

**Bit 2 to 7: reserved**

### 5.3.12 Coding of the field Diagnosis_User_Data

One of the following data types shall be used for each Diagnosis_User_Data field:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.3.13 Coding of the field Modul_Status_Array

The field Modul_Status_Array is an array of octets that shall contain at least one and at most 61 octets referred to as Modul_Status_Octet. A Modul_Status_Octet shall consist of at least one and at most 4 module stati referred to as Modul_Status_Entry_1 to Modul_Status_Entry_4 coded in two bits each. Therefore, the number of Modul_Status_Octet corresponds to the number of configured modules within a device and shall be calculated as follows

a) $N = 1$ if $M_{highest} \leq 4$

b) $N = M_{highest}$ DIV 4 +1 if $M_{highest}$ MOD 4 $\neq 0$

c) $N = M_{highest}$ DIV 4 if $M_{highest}$ MOD 4 = 0

where

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

N is the number of octets Modul_Status_Octet or the number of array elements,

$M_{highest}$ is the highest number of configured modules within a device (maximum 244).

The last Modul_Status_Octet (octet N) may not contain all 4 module status entries. If $M_{highest}$ MOD 4 ≠0 the octet N is not fully filled and the remaining bits shall be set to zero referred to as Padding Bits.

The stati of the device's modules shall be structured in ascending order without gaps. The status of module number one is always placed in Modul_Status_Entry_1 of the Modul_Status_Octet number one. The coding of a Modul_Status_Octet shall be according to 3.4 and the individual bits shall have the following meaning:

NOTE The term configured modules" addresses physical or virtual modules of a device that have had a related format field in the Chk_Cfg-REQ-PDU. Modules that are not part of the configuration are purely local and therefore not related to the Modul_Status_Array field.

**Bit 0 to 1: Modul_Status_Entry_1**
It shall contain the module status of a configured module with the number that meets m MOD 4 = 1. Padding bits shall not be used here.

**Bit 2 to 3: Modul_Status_Entry_2**
It shall contain the module status of a configured module with the number that meets m MOD 4 = 2. Otherwise padding bits (00 binary) shall be used.

**Bit 4 to 5: Modul_Status_Entry_3**
It shall contain the module status of a configured module with the number that meets m MOD 4 = 3. Otherwise padding bits (00 binary) shall be used.

**Bit 6 to 7: Modul_Status_Entry_4**
These 2 bits shall contain the module status of a configured module with the number that meets m MOD 4 = 0. Otherwise padding bits (00 binary) shall be used.

where
m is the number of the current module with 1 ≤ m ≤ N.

Figure 7 illustrates the arrangement of Modul_Status_Entry_(1 to 4) fields as an example for a device with 6 configured modules.

| | **Bit Number** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Modul_ Status_ Octet 1 | status module number 4 | | status module number 3 | | status module number 2 | | status module number 1 | |
| Modul_ Status_ Octet 2 | Padding Bits | | Padding Bits | | status module number 6 | | status module number 5 | |
| | Modul_Status_Entry_4 | | Modul_Status_Entry_3 | | Modul_Status_Entry_2 | | Modul_Status_Entry_1 | |

**Figure 7 – Example Modul_Status_Array**

Each entry Modul_Status_Entry_1 to Modul_Status_Entry_4 shall contain the module status according to Table 9.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 9 – Range of Modul_Status_Entry (1-4)**

| Value (decimal) | Meaning |
|---|---|
| 0 | data valid |
| 1 | data invalid: the data of the corresponding module are not valid due to an error (e.g. short circuit) |
| 2 | data invalid/wrong module: the data of the corresponding module are not valid, due to a wrong module in place |
| 3 | data invalid/no module: the data of the corresponding module are not valid, because there is no module in place |

### 5.3.14 Coding of the field Identifier_Diagnosis_Data_Array

The field Identifier_Diagnosis_Data_Array is an array of octets that shall contain at least one and at most 31 octets referred to as Identifier_Diagnosis_Data_Octet. A Identifier_Diagnosis_Data_Octet shall consist of at least one and at most 8 diagnosis entries referred to as Identifier_Diagnosis_Entry_1 to Identifier_Diagnosis_Entry_8. Therefore, the number of Identifier_Diagnosis_Data_Octet corresponds to the number of configured modules within a device and shall be calculated as follows

a) N = 1 if $M_{highest} \leq 8$

b) N = $M_{highest}$ DIV 8 +1 if $M_{highest}$ MOD 4 $\neq$ 0

c) N = $M_{highest}$ DIV 8 if $M_{highest}$ MOD 4 = 0

where

N is the number of octets Identifier_Diagnosis_Data_Octet or the number of array elements,

$M_{highest}$ is the highest number of configured modules within a device (maximum 244).

The last Identifier_Diagnosis_Data_Octet (octet N) may not contain all 8 diagnosis entries. If $M_{highest}$ MOD 8 $\neq$ 0 the octet N is not fully filled and the remaining bits shall be set to zero referred to as Padding Bits.

NOTE The term DIV stands for division without rest. The term MOD stands for the rest of the division.

The identifier diagnosis of the device's modules shall be structured in ascending order without gaps. The identifier diagnosis of module number one is always placed in Identifier_Diagnosis_Entry_1 of the Identifier_Diagnosis_Data_Octet number one. The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: Identifier_Diagnosis_Entry_1**
This bit shall be set to one if the module with the number that meets m MOD 8 = 1 indicates diagnosis otherwise it shall be set to zero. A padding bit shall not be used here.

**Bit 1: Identifier_Diagnosis_Entry_2**
This bit shall be set to one if the module with the number that meets m MOD 8 = 2 indicates diagnosis otherwise it shall be set to zero. A padding bit shall be used if there is no module configured.

**Bit 2: Identifier_Diagnosis_Entry_3**
This bit shall be set to one if the module with the number that meets m MOD 8 = 3 indicates diagnosis otherwise it shall be set to zero. A padding bit shall be used if there is no module configured.

**Bit 3: Identifier_Diagnosis_Entry_4**
This bit shall be set to one if the module with the number that meets m MOD 8 = 4 indicates diagnosis otherwise it shall be set to zero. A padding bit shall be used if there is no module configured.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Bit 4: Identifier_Diagnosis_Entry _5**
This bit shall be set to one if the module with the number that meets m MOD 8 = 5 indicates diagnosis otherwise it shall be set to zero. A padding bit shall be used if there is no module configured.

**Bit 5: Identifier_Diagnosis_Entry _6**
This bit shall be set to one if the module with the number that meets m MOD 8 = 6 indicates diagnosis otherwise it shall be set to zero. A padding bit shall be used if there is no module configured.

**Bit 6: Identifier_Diagnosis_Entry _7**
This bit shall be set to one if the module with the number that meets m MOD 8 = 7 indicates diagnosis otherwise it shall be set to zero. A padding bit shall be used if there is no module configured.

**Bit 7: Identifier_Diagnosis_Entry_8**
This bit shall be set to one if the module with the number that meets m MOD 8 = 0 indicates diagnosis otherwise it shall be set to zero. A padding bit shall be used if there is no module configured.

where m is the number of the current module with $1 \leq m \leq N$.

### 5.3.15 Coding of the field Identifier_Number

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 5: Identifier_Number**
The values zero to 63 (decimal) are valid.

**Bit 6 to 7: Selection**
**Decimal (2): means Channel Related Diagnosis**

### 5.3.16 Coding of the field Channel_Number

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 5: Channel_Number**
The values zero to 63 (decimal) are valid.

**Bit 6 to 7: Input_Output_Selection**
Decimal (0): reserved

Decimal (1): means input

Decimal (2): means output

Decimal (3): means input and output

### 5.3.17 Coding of the field Type_of_Diagnosis

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 4: Error_Type**
The Error_Type shall be set as shown in Table 10.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 10 – Error type**

| Value (decimal) | Meaning |
|---|---|
| 0 | reserved |
| 1 | short circuit |
| 2 | Undervoltage |
| 3 | Overvoltage |
| 4 | Overload |
| 5 | Overtemperature |
| 6 | line break |
| 7 | upper limit value exceeded |
| 8 | lower limit value exceeded |
| 9 | Error |
| 10 – 31 | See ChannelErrorType in IEC 61158-6-10, 6.2.5.2 |

**Bit 5 to 7: Channel_Type**
Decimal (0): unspecific, may be used for any type

Decimal (1): means bit

Decimal (2): means 2 bit

Decimal (3): means 4 bit

Decimal (4): means octet

Decimal (5): means word

Decimal (6): means 2 words

Decimal (7): reserved

### 5.3.18 Coding of the field Revision_Number

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 5: Revision_Number**
The values one to 63 (decimal) are valid.

**Bit 6: shall be set to one**

**Bit 7: shall be set to one**

### 5.3.19 Coding of the field Publisher_Address

This field shall be coded as data type Unsigned8. The range of values shall be zero to 125. This parameter shall contain the address of the DP-slave acting as Publisher.

### 5.3.20 Coding of the field Publisher_Status

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 7: Link_Status**
TRUE (1) means no error and active, FALSE (0) error or not active during the last monitoring period

**Bit 6: Link_Error**
TRUE (1) means length error, FALSE (0) means no error

**Bit 0 to 5: reserved**

### 5.3.21 Coding of the field RedSpecifier

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 1: Status_Specifier**
Decimal (0): means no further differentiation

**Bit 3 to 7: Seq Number**
Contains Sequence Number of the last PrmCmd processed

### 5.3.22 Coding of the field Function

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: reserved**
**Bit 1: Primary**
TRUE (1) means a Primary request was executed last, FALSE (0) means no Primary request was executed last

**Bit 2: Start MS1**
TRUE (1) means a Start MS1 request was executed last, FALSE (0) means no Start MS1 request was executed last

**Bit 3: Stop MS1**
TRUE (1) means a Stop MS1 request was executed last, FALSE (0) means no Stop MS1 request was executed last

**Bit 4: Check_Properties**
TRUE (1) means a Check_Properties request was executed last, FALSE (0) means no Check_Properties request was executed last

**Bit 5: reserved**
**Bit 6: MasterStateClear**
TRUE (1) means a MasterStateClear request was executed last, FALSE (0) means no MasterStateClear request was executed last

**Bit 7: reserved**

### 5.3.23 Coding of the field Red_Status1

This field represents the Status of the Component which sent the status. The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: Backup**
TRUE (1) means Component is in Backup State, FALSE (0) means Component is not in Backup State

**Bit 1: Primary**
TRUE (1) means Component is in Primary State, FALSE (0) means Component is not in Primary State

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Bit 2: HW-Defect**
TRUE (1) means Component has a Hardware Defect, FALSE (0) means has no Hardware Defect

**Bit 3: DataExchange**
TRUE (1) means Component is in DataExchange State, FALSE (0) means Component is not in DataExchange State

**Bit 4: Fail Safe**
TRUE (1) means Component is in Fail Safe State, FALSE (0) means Component is not in Fail Safe State

**Bit 5: DatarateSet**
TRUE (1) means Component has a Datarate set, FALSE (0) means has no Datarate set

**Bit 6: TohStarted**
TRUE (1) means Component has a started Toh, FALSE (0) means has not started Toh

**Bit 7: reserved**

### 5.3.24 Coding of the field Red_Status2

This field represents the Status of a Partner Component of the Component which sent the status. The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: Backup**
TRUE (1) means Component is in Backup State, FALSE (0) means Component is not in Backup State

**Bit 1: Primary**
TRUE (1) means Component is in Primary State, FALSE (0) means Component is not in Primary State

**Bit 2: HW-Defect**
TRUE (1) means Component has a Hardware Defect, FALSE (0) means has no Hardware Defect

**Bit 3: DataExchange**
TRUE (1) means Component is in DataExchange State, FALSE (0) means Component is not in DataExchange State

**Bit 4: Fail Safe**
TRUE (1) means Component is in Fail Safe State, FALSE (0) means Component is not in Fail Safe State

**Bit 5: DatarateSet**
TRUE (1) means Component has a Datarate set, FALSE (0) means has no Datarate set

**Bit 6: TohStarted**
TRUE (1) means Component has a started Toh, FALSE (0) means has not started Toh

**Bit 7: reserved**

### 5.3.25 Coding of the field Red_Status3

This field shall be coded as data type Unsigned8. The values are user specific.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

## 5.4 Coding section related to parameterisation PDU

### 5.4.1 Coding of the field Station_status

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: reserved**

**Bit 1: reserved**

**Bit 2: reserved**

**Bit 3: WD_On (Watchdog on)**
FALSE (0), TRUE (1)

**Bit 4: Freeze_Req**
Value(0): Freeze mode not requested

Value(1): Freeze mode requested

**Bit 5: Sync_Req**
Value(0): Sync mode not requested

**Bit 6: Unlock_Req**
Shall be used as described in Table 11.

Value(1): Sync mode requested

**Bit 7: Lock_Req**
Shall be used as described in Table 11.

**Table 11 – Specification of the bits Lock_Req and Unlock_Req**

| Bit 7 | Bit 6 | Meaning |
|---|---|---|
| 0 | 0 | The parameter min $T_{SDR}$ can be changed. All other parameters remain unchanged. |
| 0 | 1 | The DP-slave will be unlocked for other Masters. |
| 1 | 0 | The DP-slave is locked for other Masters, all parameters are accepted (exception: min $T_{SDR}$ = 0) |
| 1 | 1 | The DP-slave is unlocked for other Masters. |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.4.2 Coding of the field WD_Fact_1

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 1 to 255

### 5.4.3 Coding of the field WD_Fact_2

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 1 to 255

### 5.4.4 Coding of the field min_$T_{SDR}$

This field shall be coded as data type Unsigned8 with the following value range:

Decimal(0 to max $T_{SDR}$ (see Part 4)) if DP operation

Decimal(0): means current value remains unchanged

### 5.4.5 Coding of the field Group_Ident

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

Bit 0: Group_1

Bit 1: Group_2

Bit 2: Group_3

Bit 3: Group_4

Bit 4: Group_5

Bit 5: Group_6

Bit 6: Group_7

Bit 7: Group_8

Each of the bits above shall set to one if the device belongs to the indicated group number. Otherwise it shall be set to zero.

NOTE A device may belong to no groups (Group_Ident=decimal(0)) or all groups (Group_Ident=decimal(255)).

### 5.4.6 Coding of the field User_Prm_Data_Element

One of the following data types shall be used for each User_Prm_Data_Element:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.4.7 Coding of the field DPV1_Status_1

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: reserved**

**Bit 1: reserved**

**Bit 2: WD_Base_1ms**
Value(0): Watchdog time base equals 10 milliseconds

Value(1): Watchdog time base equals 1 millisecond

**Bit 3 to 4: reserved**

**Bit 5: Publisher_Enable**
FALSE(0), TRUE(1)

**Bit 6: Fail_Safe**
FALSE(0), TRUE(1)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 7: DPV1_Enable**
FALSE(0), TRUE(1)

### 5.4.8 Coding of the field DPV1_Status_2

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: Check_Cfg_Mode**
Value(0): normal check (restrictive)

Value(1): user specific check (more flexible)

**Bit 1: reserved**

**Bit 2: Enable_Update_Alarm**
FALSE(0), TRUE(1)

**Bit 3: Enable_Status_Alarm**
FALSE(0), TRUE(1)

**Bit 4: Enable_Manufacturer_Specific_Alarm**
FALSE(0), TRUE(1)

**Bit 5: Enable_Diagnostic_Alarm**
FALSE(0), TRUE(1)

**Bit 6: Enable_Process_Alarm**
FALSE(0), TRUE(1)

**Bit 7: Enable_Pull_Plug_Alarm**
FALSE(0), TRUE(1)

### 5.4.9 Coding of the field DPV1_Status_3

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 2: Alarm_Mode**
Decimal(0): 1 alarm of each type

Decimal(1): 2 alarms in total

Decimal(2): 4 alarms in total

Decimal(3): 8 alarms in total

Decimal(4): 12 alarms in total

Decimal(5): 16 alarms in total

Decimal(6): 24 alarms in total

Decimal(7): 32 alarms in total

**Bit 3: Prm_Structure**
FALSE(0), TRUE(1)

**Bit 4: IsoM_Req**
FALSE(0), TRUE(1)

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Bit 5 to 6: reserved**
**Bit 7: PrmCmd**
FALSE(0), TRUE(1)

### 5.4.10 Coding of the field Structure_Length

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 5 to 244, 255

The field Structure_Length contains the number of octets including itself. A value of 255 means that all elements in the Prm_User_Data following this element belongs to this structure.

### 5.4.11 Coding of the field Structure_Type

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 0 to 1: reserved

Decimal 2: PrmCmd

Decimal 3: DXB Linktable

Decimal 4: IsoM Parameter

Decimal 5: F-Parameter

Decimal 6: reserved

Decimal 7: DXB Subscribertable

Decimal 8: Time AR Parameter

Decimal 9 to 31: reserved

Decimal 32 to 128: Manufacturer Specific

Decimal 129: User Prm Data

Decimal 130 to 255: reserved

### 5.4.12 Coding of the field Version

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 0: reserved

Decimal 1: Version 1

Decimal 2 to 255: reserved

### 5.4.13 Coding of the field Publisher_Addr

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 0 to 125, 128

Decimal 126, 127, 129 to 255: reserved

In case of the DXB-Subscribertable the value 128 indicates an entry for the DP-master (Class 1).

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.4.14 Coding of the field Publisher_Length

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 1 to 244

Decimal 0, 245 to 255: reserved

In case of a Master Data entry in the DXB-Subscribertable the value in this field is not relevant and shall be set to 0.

#### 5.4.15 Coding of the field Sample_Offset

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 0 to 243

Decimal 244 to 255: reserved

#### 5.4.16 Coding of the field Sample_Length

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 1 to 244

Decimal 0, 245 to 255: reserved

#### 5.4.17 Coding of the Dest_Slot_Number

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 1 to 244

Decimal 0, 245 to 255: reserved

#### 5.4.18 Coding of the Offset_Data_Area

This field shall be coded as data type Unsigned8 with the following value range:

Decimal 0 to 244

Decimal 245 to 255: reserved

#### 5.4.19 Coding of the field $T_{BASE_DP}$

This field shall be coded as data type Unsigned32 with the allowed values of decimal 375, 750, 1500 (default), 3000, 6000, 12000. All other values are reserved.

#### 5.4.20 Coding of the field $T_{DP}$

This field shall be coded as data type Unsigned16 with the value range from 154 to $2^{16}$-1.

#### 5.4.21 Coding of the field $T_{MAPC}$

This field shall be coded as data type Unsigned8.

#### 5.4.22 Coding of the field $T_{BASE_IO}$

This field shall be coded as data type Unsigned32 with the allowed values of decimal 375, 750, 1500 (default), 3000, 6000, 12000. All other values are reserved.

#### 5.4.23 Coding of the field $T_I$

This field shall be coded as data type Unsigned16.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.4.24 Coding of the field $T_O$

This field shall be coded as data type Unsigned16.

#### 5.4.25 Coding of the field $T_{DX}$

This field shall be coded as data type Unsigned32.

#### 5.4.26 Coding of the field $T_{PLL_W}$

This field shall be coded as data type Unsigned16 with the value range from 1 to $2^{16}$-1.

#### 5.4.27 Coding of the field $T_{PLL_D}$

This field shall be coded as data type Unsigned16 with the value range from 0 to $2^{16}$ –1.

#### 5.4.28 Coding of the field Specifier

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 1: Specifier**
Decimal (0): means no further differentiation

**Bit 3 to 7: Seq Number**
Contains Sequence Number of the PrmCmd

#### 5.4.29 Coding of the field Function

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: reserved**
**Bit 1: Primary**
TRUE (1) means a Primary request is requested, FALSE (0) means no Primary request is requested

**Bit 2: Start MS1**
TRUE (1) means a Start MS1 request is requested, FALSE (0) means no Start MS1 request is requested

**Bit 3: Stop MS1**
TRUE (1) means a Stop MS1 request is requested, FALSE (0) means no Stop MS1 request is requested

**Bit 4: Check_Properties**
TRUE (1) means a Check_Properties request is requested, FALSE (0) means no Check_Properties request is requested

**Bit 5: reserved**
**Bit 6: MasterStateClear**
TRUE (1) means the MasterState is Clear, FALSE (0) means the MasterState is Operate

**Bit 7: reserved**

#### 5.4.30 Coding of the field Properties

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 0: Primary**
TRUE (1) means a Primary request is used, FALSE (0) means no Primary request is used

**Bit 1: Start Stop MS1**
TRUE (1) means a Start Stop MS1 request is used, FALSE (0) means no Start Stop MS1 request is used

**Bit 2: AddressChange**
TRUE (1) means Address Change is required with Primary/Backup role, FALSE (0) means no Address Change is required with Primary/Backup role

**Bit 3: AddressOffset**
TRUE (1) means Address Offset is 64, FALSE (0) means Address Offset is 0.

**Bit 4 to 7: reserved**

#### 5.4.31 Coding of the field Output Hold Time

This field shall be coded as data type Unsigned16 with the value range from 0 to $2^{16}$-1.

#### 5.4.32 Coding of the field Clock Sync Interval

This field shall be coded as data type Unsigned16 with the value range from 0 to $2^{16}$-1.

#### 5.4.33 Coding of the field CS Delay Time

This field shall be coded as data type Network Time Difference.

### 5.5 Coding section related to configurationPDUs

#### 5.5.1 Coding of the field Cfg_Identifier

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 3: Data_Length**
Decimal(0): means 1 octet or 1 word according to the Length_Format

Decimal(1): means 2 octets or 2 words according to the Length_Format

Decimal(2): means 3 octets or 3 words according to the Length_Format

Decimal(3): means 4 octets or 4 words according to the Length_Format

Decimal(4): means 5 octets or 5 words according to the Length_Format

…

Decimal(15): means 16 octets or 16 words according to the Length_Format

**Bit 4 to 5: Input_Output_Selection**
Decimal(0): shall not be used here, means specific identifier format

Decimal(1): means input

Decimal(2): means output

Decimal(3): means input and output

**Bit 6: Length_Format**
Value(0) : means octet structure used

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Value(1) : means word structure used (high octet transferred first, big-endian)

**Bit 7: Consistency**
Value(0): means octet or word consistency

Value(1): means consistency over the whole length

### 5.5.2 Coding of the field Special_Cfg_Identifier

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 3: Length_of_Manufacturer_Specific_Data**
The length information of the manufacturer specific data shall be in dependency of the use in the Chk_Cfg-REQ-PDU or Get_Cfg-RES-PDU as shown in Table 12 and Table 13. Data type fields shall always be counted if present.

**Bit 4 to 5: shall be set to zero (decimal)**

**Bit 6 to 7: Input_Output_Selection**
Decimal(0): means empty slot, no input or output data for this module

Decimal(1): means one length octet for input data follows

Decimal(2): means one length octet for output data follows

Decimal(3): means one length octet for output data followed by one length octet for input data follows

**Table 12 – Range of Length_of_Manufacturer_Specific_Data if used in Chk_Cfg-REQ-PDU**

| Value (decimal) | Meaning |
|---|---|
| 0 | No manufacturer specific data follow; no data in Real_Cfg_Data. |
| 1 to 14 | Manufacturer specific data of specified length follow; these shall be identical with the data in Real_Cfg_Data. |
| 15 | No manufacturer specific data follow; the verification can be omitted. |

**Table 13 – Range of Length_of_Manufacturer_Specific_Data if used in Get_Cfg-RES-PDU**

| Value (decimal) | Meaning |
|---|---|
| 0 | No manufacturer specific data follow. |
| 1 to 14 | Manufacturer specific data with specified length follow. |
| 15 | Not allowed. |

### 5.5.3 Coding of the fields Length_Octet

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 5: Data_Length**
Decimal(0): means 1 octet or 1 word according to the Length_Format

Decimal(1): means 2 octets or 2 words according to the Length_Format

...

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Decimal(63): means 64 octets or 64 words according to the Length_Format

**Bit 6: Length_Format**
Value(0): means octet structure used

Value(1): means word structure used (high octet transferred first, big-endian)

**Bit 7: Consistency**
Value(0): means octet or word consistency

Value(1): means consistency over the whole length indicated in Data_Length

### 5.5.4 Coding of the field Manufacturer_Specific_Data

One of the following data types shall be used for each Manufacturer_Specific_Data field:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.5.5 Coding of the field Extended_Length_Octet

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 5: Data_Length**
Decimal(0): means 1 octet

Decimal(1): means 2 octets

Decimal(2): means 3 octets

Decimal(3): means 4 octets

Decimal(4): means 5 octets

...

Decimal(63): means 64 octets

**Bit 6: Format**
This bit shall be set to zero that means octet structure is used.

**Bit 7: Consistency**
This bit shall be set to one that means consistency over the whole length.

### 5.5.6 Coding of the field Data_Type

This field shall be coded as data type Unsigned8 with values according to the code column at Table 14.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 14 – Values (codes) for data types**

| Data type name | Code (decimal) | Number of octets for the data type |
|---|---|---|
| Boolean | 1 | 1 |
| Integer8 | 2 | 1 |
| Integer16 | 3 | 2 |
| Integer32 | 4 | 4 |
| Unsigned8 | 5 | 1 |
| Unsigned16 | 6 | 2 |
| Unsigned32 | 7 | 4 |
| Floating Point 32 | 8 | 4 |
| Visible String | 9 | 1, 2, 3, … |
| Octet String | 10 | 1, 2, 3, … |
| — | 11 | reserved |
| TimeOfDay (with Date indication) | 12 | 6 |
| — | 13 to 14 | reserved |
| Float64 | 15 | 8 |
| — | 16 to 49, 51 | reserved |
| Date | 50 | 7 |
| TimeOfDay (without Date indication) | 52 | 4 |
| Time Difference (with Date indication) | 53 | 6 |
| Time Difference (without Date indication) | 54 | 4 |
| Integer64 | 55 | 8 |
| Unsigned64 | 56 | 8 |
| — | 57 | reserved |
| Network Time | 58 | 8 |
| Network Time Difference | 59 | 8 |
| — | 60 to 70 | reserved |
| — | 71 to 100 | User specific |
| — | 101 to 127 | reserved |
| — | 128 to 255 | User specific (no data type coding) |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Sequences of data types shall be coded as a list of simple data types. The sum of the lengths of the data types shall be corresponding the input- or output data length. Data types with variable length (visible string, octet string, time of day, time difference) shall not be handled in a sequence, but as single element.

The user specific coding (value>=128) shall not be used in any combination with type specific coding (value<=70) in one Chk_Cfg-REQ-PDU or Get_Cfg-REQ-PDU.

### 5.6 Coding section related to global control PDUs

#### 5.6.1 Coding of the field Control_Command

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: reserved**

**Bit 1: Clear_Data**
Value(0): no command

Value(1): clear output (set output data in safe state)

**Bit 2: UnFreeze**
Value(0): no command

Value(1): cancels Freeze command

**Bit 3: Freeze**
Value(0): no command

Value(1): freeze input values

**Bit 4: Unsync**
Value(0): no command

Value(1): cancels Sync command

**Bit 5: Sync**
Value(0): no command

Value(1): synchronise output values

**Bit 6: reserved**

**Bit 7: reserved**
Table 15 illustrates the meaning and priority of the Bits 2 to 5.

**Table 15 – Specification of the bits for Un-/Sync and Un-/Freeze**

| Bit 2 resp. 4 | Bit 3 resp. 5 | Meaning |
|---|---|---|
| 0 | 0 | no function |
| 0 | 1 | function is activated |
| 1 | 0 | function is deactivated |
| 1 | 1 | function is deactivated |

#### 5.6.2 Coding of the field Group_Select

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

Bit 0: Group_1

Bit 1: Group_2

Bit 2: Group_3

Bit 3: Group_4

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Bit 4: Group_5

Bit 5: Group_6

Bit 6: Group_7

Bit 7: Group_8

Each of the bits above shall set to one if the control command belongs to the indicated group number. Otherwise it shall be set to zero. Zero means that all assigned Slaves shall execute the Control Command.

NOTE A control command may belong to no groups (Group_Ident=decimal(0)) or all groups (Group_Ident=decimal(255)).

### 5.7 Coding section related to clock-value-PDUs

#### 5.7.1 Coding of the field Clock_value_time_event

The coding of this field shall be as data type Network Time with the fixed length of 8.

#### 5.7.2 Clock_value_previous_TE

The coding of this field shall be as data type Network Time with the fixed length of 8.

#### 5.7.3 Coding of the field Clock_value_status1

The coding of this field shall be according to 3.4, with C and CV mapped to the parameter local time diff in the set time service. The individual bits shall have the following meaning:

**Bit 0: reserved**

**Bit 1: reserved**

**Bit 2 to 6: CV (Correction Value)**
Value(0): 0 min

Value(1): 30 min

Value(2): 60 min

Value(3): 90 min

...

Value(31): 930 min

**Bit 7: C (Sign of CV)**
Value(0): CV to be added from Time

Value(1): CV to be subtracted from Time

#### 5.7.4 Coding of the field Clock_value_status2

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: SYF (Synchronisation fault)**
Value(0): Clock_value_time_event is synchronised

Value(1): Clock_value_time_event is not synchronised (with other clocks in the system)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 1: reserved**

**Bit 2: reserved**

**Bit 3 to 4: CR (Accuracy)**
Value(0): 1 ms

Value(1): 10 ms

Value(2): 100 ms

Value(3): 1 s

**Bit 5: reserved**

**Bit 6: SWT (Summer/Winter-Time)**
Value(0): Winter Time

Value(1): Summer Time

**Bit 7: ANH (Announcement Hour)**
Value(0): No Change planned within the next hour

Value(1): A Change of SWT will occur within the next hour

### 5.8 Coding section related to function identification and errors

#### 5.8.1 Coding of the field Function_Num

The coding of this field shall be according to 3.4 and the individual bits shall have the following meanings. The meaning of the complete field is defined in Table 16.

**Bit 0 to 4: Function_Code / Error_Code**
This bits contain the Function_Code or the Error_Code corresponding to the DLPDU_Selector. The Function_Code shall be set according to Table 16.

**Bit 5 to 6: PDU_Identifier**
These two bits shall be set to 2 (decimal) to indicate a DP-PDU.

**Bit 7: DLPDU_Selector**
Value(0): means request DLPDU or positive response DLPDU

Value(1): means error or negative response DLPDU

The field Function_Num is transferred in the request and response DLPDU. If the service is processed correctly, bit 7 of Function_Num in the response APDU shall be set to zero. In case of an error an error DLPDU is transferred. In this APDU Function_Num contains an error code and the bit 7 shall be set to one.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 16 – Coding of the Function_Code/ Function_Num**

| DLPDU_ Selector (decimal) | PDU_ Identifier (decimal) | Function_Code (decimal) | Meaning | Complete field: Function_Num (hexadecimal) |
|---|---|---|---|---|
| 0 | 2 | 0 | reserved | 0x40 |
| 0 | 2 | 1 | Get_Master_Diag | 0x41 |
| 0 | 2 | 2 | Start_Seq | 0x42 |
| 0 | 2 | 3 | Download | 0x43 |
| 0 | 2 | 4 | Upload | 0x44 |
| 0 | 2 | 5 | End_Seq | 0x45 |
| 0 | 2 | 6 | Act_Para_Brct | 0x46 |
| 0 | 2 | 7 | Act_Param | 0x47 |
| 0 | 2 | 8 | Idle | 0x48 |
| 0 | 2 | 9 to 16 | reserved | 0x49 to 0x50 |
| 0 | 2 | 17 | Data_Transport | 0x51 |
| 0 | 2 | 18 to 21 | reserved | 0x52 to 0x55 |
| 0 | 2 | 22 | RM | 0x56 |
| 0 | 2 | 23 | Initiate | 0x57 |
| 0 | 2 | 24 | Abort | 0x58 |
| 0 | 2 | 25 | reserved | 0x59 |
| 0 | 2 | 26 | reserved | 0x5A |
| 0 | 2 | 27 | reserved | 0x5B |
| 0 | 2 | 28 | Alarm_Ack | 0x5C |
| 0 | 2 | 29 | reserved | 0x5D |
| 0 | 2 | 30 | Read | 0x5E |
| 0 | 2 | 31 | Write | 0x5F |

The Error_Code shall be set according to Table 17.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 17 – Coding of the Error_Code / Function_Num**

| DLPDU_ Selector (decimal) | PDU_ Identifier (decimal) | Error_Code (decimal) | Meaning | Complete field: Function_Num (hexadecimal) |
|---|---|---|---|---|
| 1 | 2 | 0 | reserved | 0xC0 |
| 1 | 2 | 1 | FE | 0xC1 |
| 1 | 2 | 2 | NI | 0xC2 |
| 1 | 2 | 3 | AD | 0xC3 |
| 1 | 2 | 4 | EA | 0xC4 |
| 1 | 2 | 5 | LE | 0xC5 |
| 1 | 2 | 6 | RE | 0xC6 |
| 1 | 2 | 7 | IP | 0xC7 |
| 1 | 2 | 8 | SC | 0xC8 |
| 1 | 2 | 9 | SE | 0xC9 |
| 1 | 2 | 10 | NE | 0xCA |
| 1 | 2 | 11 | DI | 0xCB |
| 1 | 2 | 12 | NC | 0xCC |
| 1 | 2 | 13 | TO | 0xCD |
| 1 | 2 | 14 | CA | 0xCE |
| 1 | 2 | 15 to 16 | reserved | 0xCF to 0xD0 |
| 1 | 2 | 17 | Error Data_Transport | 0xD1 |
| 1 | 2 | 18 to 22 | reserved | 0xD2 to 0xD6 |
| 1 | 2 | 23 | Error Initiate | 0xD7 |
| 1 | 2 | 24 | reserved | 0xD8 |
| 1 | 2 | 25 | reserved | 0xD9 |
| 1 | 2 | 26 | reserved | 0xDA |
| 1 | 2 | 27 | reserved | 0xDB |
| 1 | 2 | 28 | Error Alarm_Ack | 0xDC |
| 1 | 2 | 29 | reserved | 0xDD |
| 1 | 2 | 30 | Error Read | 0xDE |
| 1 | 2 | 31 | Error Write | 0xDF |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.8.2 Coding of the field Error_Decode

The coding of this field shall be as data type Unsigned8 and the values shall be set according to Table 18.

**Table 18 – Values of Error_Decode**

| Value(decimal) | Meaning |
|---|---|
| 0 to 127 | reserved |
| 128 = | DPV1 |
| 129 to 253 = | reserved |
| 254 to 255 = | PROFILE_SPECIFIC |

The field Error_Decode defines the meaning of the following Octets Error_Code_x. The Error_Decode values PROFILE_SPECIFIC indicate that the parameters Error_Code_x shall be set as defined in the corresponding protocols:

PROFILE_SPECIFIC

Error_Code_1= taken from profile specification

Error_Code_2= taken from profile specification

### 5.8.3 Coding of the field Error_Code_1

If field Error_Decode = DPV1

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 3: Error_Code**
The values shall be set according to the Error_Code column in Table 19.

**Bit 4 to 7: Error_Class**
The values shall be set according to the Error_Class column in Table 19.

**Table 19 – Coding of Error_Code_1 at DPV1**

| Error_Class (decimal) | Meaning | Error_Code (decimal) |
|---|---|---|
| 0 to 9 | not specified | not specified *) |
| 10 | application | 0 = read error<br>1 = write error<br>2 = module failure<br>3 to 6 = not specified<br>7 = busy<br>8 = version conflict<br>9 = feature not supported<br>10 to 15 = User specific |
| 11 | access | 0 = invalid index<br>1 = write length error<br>2 = invalid slot<br>3 = type conflict<br>4 = invalid area<br>5 = state conflict<br>6 = access denied<br>7 = invalid range<br>8 = invalid parameter<br>9 = invalid type<br>10= backup<br>11 to 15 = User specific |
| 12 | resource | 0 = read constrain conflict<br>1 = write constrain conflict<br>2 = resource busy<br>3 = resource unavailable<br>4 to 7 = not specified<br>8 to 15 = User specific |
| 13 to 15 | User specific | |

NOTE Not specified values are used to serve legacy codes and are intended to be passed unchanged to the application.

If field Error_Decode = PROFILE_SPECIFIC

Error_Code_1= from profile specification, needs to be specified in the profile definition

### 5.8.4 Coding of the field Error_Code_2

If field Error_Decode = DPV1

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

The coding of this field shall be as data type Unsigned8. The values are user specific.

If field Error_Decode = PROFILE_SPECIFIC

Error_Code_2 = from profile specification, needs to be specified in the profile definition.

## 5.9 Coding section related to master diagnosis PDU

### 5.9.1 Coding of the field MDiag_Identifier

The coding of this field shall be as data type Unsigned8 and the values shall be set according Table 20:

**Table 20 – Values of MDiag_Identifier**

| Value (decimal) | Meaning |
|---|---|
| 0-125 | Diag_Data of the DP-slave |
| 126 | System_Diagnostis |
| 127 | Master_Status |
| 128 | Data_Transfer_List |
| 129-255 | reserved |

### 5.9.2 Coding of the field System_Diagnosis

The coding of this field shall be as data type Octet String with the fixed length of 16. The 16 octets of System_Diagnosis shall be coded as follows

**Octet 1**
Bit 0 shall be set to one if station number 0 has reported diagnosis otherwise it shall be set to zero.

Bit 1 shall be set to one if station number 1 has reported diagnosis otherwise it shall be set to zero.

Bit 2 shall be set to one if station number 2 has reported diagnosis otherwise it shall be set to zero.

Bit 3 shall be set to one if station number 3 has reported diagnosis otherwise it shall be set to zero.

Bit 4 shall be set to one if station number 4 has reported diagnosis otherwise it shall be set to zero.

Bit 5 shall be set to one if station number 5 has reported diagnosis otherwise it shall be set to zero.

Bit 6 shall be set to one if station number 6 has reported diagnosis otherwise it shall be set to zero.

Bit 7 shall be set to one if station number 7 has reported diagnosis otherwise it shall be set to zero.

**Octet 2**
Bit 0 shall be set to one if station number 8 has reported diagnosis otherwise it shall be set to zero.

Bit 1 shall be set to one if station number 9 has reported diagnosis otherwise it shall be set to zero.

etc.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Octet 16**

.

.

Bit 5 shall be set to one if station number 125 has reported diagnosis otherwise it shall be set to zero.

Bit 6 to 7 shall be set to zero (not used)

### 5.9.3 Coding of the field USIF_State

This field shall be coded as data type Unsigned8. The following values are defined:

Value(0x40): shall be set if operation mode of DP-master (Class 1) is STOP

Value(0x80): shall be set if operation mode of DP-master (Class 1) is CLEAR

Value(0xC0): shall be set if operation mode of DP-master (Class 1) is OPERATE

### 5.9.4 Coding of the field Hardware_Release_DP

This field shall be coded as data type Unsigned8.

### 5.9.5 Coding of the field Firmware Release_DP

This field shall be coded as data type Unsigned8.

### 5.9.6 Coding of the field Hardware_Release_User

This field shall be coded as data type Unsigned8.

### 5.9.7 Coding of the field Firmware Release_User

This field shall be coded as data type Unsigned8.

### 5.9.8 Coding of the field Data_Transfer_List

The coding of this field shall be as data type Octet String with the fixed length of 16. The 16 octets of Data_Transfer_List shall be coded as follows

**Octet 1**

Bit 0 shall be set to one if data exchange executed with station number 0 otherwise it shall be set to zero.

Bit 1 shall be set to one if data exchange executed with station number 1 otherwise it shall be set to zero.

Bit 2 shall be set to one if data exchange executed with station number 2 otherwise it shall be set to zero.

Bit 3 shall be set to one if data exchange executed with station number 3 otherwise it shall be set to zero.

Bit 4 shall be set to one if data exchange executed with station number 4 otherwise it shall be set to zero.

Bit 5 shall be set to one if data exchange executed with station number 5 otherwise it shall be set to zero.

Bit 6 shall be set to one if data exchange executed with station number 6 otherwise it shall be set to zero.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Bit 7 shall be set to one if data exchange executed with station number 7 otherwise it shall be set to zero.

**Octet 2**
Bit 0 shall be set to one if data exchange executed with station number 8 otherwise it shall be set to zero.

Bit 1 shall be set to one if data exchange executed with station number 9 otherwise it shall be set to zero.

etc.

**Octet 16**
.

.

Bit 5 shall be set to one if data exchange executed with station number 125 otherwise it shall be set to zero.

Bit 6 to 7 shall be set to zero (not used)

### 5.10 Coding section related to upload/download/act para PDUs

#### 5.10.1 Coding of the field Area_Code_UpDownload

The coding of this field shall be as data type Unsigned8 and the values shall be set according to Table 21.

**Table 21 – Values for Area_Code_UpDownload**

| Value (decimal) | Meaning |
|---|---|
| 0 to 125 | DP-slave parameter set (see field DP_Slave_Parameter_Set) |
| 126 | reserved |
| 127 | Bus parameter set (see field Bus_Parameter_Set) |
| 128 | reserved |
| 129 | statistic counters (see field Statistic_Counter) |
| 130 to 135 | reserved |
| 136 to 139 | Master parameter set (see field Master_Parameter_Set) |
| 140 to 254 | reserved |
| 255 | No local access protection in Start_Seq-REQ-PDU or Start_Seq-RES-PDU |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.10.2 Coding of the field Timeout

The coding of this field shall be as data type Unsigned16 and the time base for the timer shall be 1ms.

#### 5.10.3 Coding of the field Max_Len_Data_Unit

The coding of this field shall be as data type Unsigned8 and the value for the Max_Len_Data_Unit shall be in the range from one to 240.

NOTE The minimum value for the Max_Len_Data_Unit is 48 in case of Initiate-RES_PDU.

#### 5.10.4 Coding of the field Add_Offset

The coding of this field shall be as data type Unsigned16.

### 5.10.5 Coding of the field Data

One of the following data types shall be used for each Data field:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.10.6 Coding of the field Data_Len

The coding of this field shall be as data type Unsigned8 and the value for the Data_Len shall be in the range from one to 240.

### 5.10.7 Coding of the field Area_CodeActBrct

The coding of this field shall be as data type Unsigned8 and the values shall be set according to Table 22.

**Table 22 – Values for Area_CodeActBrct**

| Value (decimal) | Meaning |
|---|---|
| 0 to 126 | reserved |
| 127 | Bus_parameter set |
| 128 to 129 | reserved |
| 130 to 135 | reserved |
| 136 to 139 | Master parameter set |
| 140 to 254 | reserved |
| 255 | reserved |

### 5.10.8 Coding of the field Area_CodeAct

The coding of this field shall be as data type Unsigned8 and the values shall be set according Table 23.

**Table 23 – Values for Area_CodeAct**

| Value (decimal) | Meaning |
|---|---|
| 0 to 125 | DP-slave parameter set<br>The Active Flag in the DP-slave parameter set of the DP-master (Class 1) is influenced accordingly. The DP-slave concerned takes part in the cyclic data exchange mode or is removed from this mode and no longer addressed. |
| 126 | reserved |
| 127 | Bus parameter set |
| 128 | operation mode |
| 129 | reserved |
| 130 to 135 | reserved |
| 136 to 139 | Master parameter set |
| 140 to 255 | reserved |

### 5.10.9 Coding of the field Activate

This field shall be coded as data type Unsigned8. The following values are defined:

Value(0x80) means activate the corresponding DP-slave parameter set if the field Area_CodeAct equals 0.. 125

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Value(0x00) means deactivate if the field Area_CodeAct equals 0.. 125

Value(0xFF) means activate the bus parameter set if the field Area_CodeAct equals 127

Value(0x40) means set operation mode to STOP if the field Area_CodeAct equals 128

Value(0x80) means set operation mode to CLEAR if the field Area_CodeAct equals 128

Value(0xC0) means set operation mode to OPERATE if the field Area_CodeAct equals 128

## 5.11 Coding section related to the bus parameter set

### 5.11.1 Coding of the field Bus_Para_Len

This field shall contain the length of Bus_Para inclusive the field Bus_Para_Len itself. This field shall be coded as data type Unsigned16. The range of values shall be from 66 to $2^{16}$-1.

### 5.11.2 Coding of the field DL_Add

This field shall be coded as data type Unsigned8. The range of values shall be zero to 125. This parameter shall contain the own address of the DP-master.

### 5.11.3 Coding of the field Data_rate

This field shall be coded as data type Unsigned8. The values shall be set as shown at Table 24:

**Table 24 – Values for Data_rate**

| Value (decimal) | Meaning |
|---|---|
| 0 | 9,6 kbit/s |
| 1 | 19,2 kbit/s |
| 2 | 93,75 kbit/s |
| 3 | 187,5 kbit/s |
| 4 | 500 kbit/s |
| 5 | reserved |
| 6 | 1 500 kbit/s |
| 7 | 3000 kbit/s |
| 8 | 6 000 kbit/s |
| 9 | 12 000 kbit/s |
| 10 | 31,25 kbit/s |
| 11 | 45,45 kbit/s |
| 12 to 255 | reserved |

### 5.11.4 Coding of the fields $T_{SL}$, min $T_{SDR}$, max $T_{SDR}$

These parameters are described in part 3. The coding of these fields shall be as data type Unsigned16.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

### 5.11.5 Coding of the fields $T_{QUI}$, $T_{SET}$, G, HSA, max_retry_limit

These parameters are described in part 3. The coding of these fields shall be as data type Unsigned8.

### 5.11.6 Coding of the field $T_{TR}$ (Target Token Rotation time)

This field is described in part 3. The coding of this field shall be as data type Unsigned32.

### 5.11.7 Coding of the field Bp_Flag (Busparameter flag)

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 2: reserved**

**Bit 3 to 4: Isochronous Mode**
Decimal (0): Not Synchronized

Decimal (1): Buffered Synchronized

Decimal (2): Enhanced Synchronized

Decimal (3): reserved

**Bit 5: IsoM Sync**
This bit shall be set to one if the DP master (Class 1) shall send a Global_Control-REQ-PDU with Control Command Sync in case of isochronous operation. Otherwise this bit shall be set to zero.

**Bit 6: IsoM Freeze**
This bit shall be set to one if the DP master (Class 1) shall send a Global_Control-REQ-PDU with Control Command Freeze in case of isochronous operation. Otherwise this bit shall be set to zero.

**Bit 7: Error_Action_Flag**
This bit shall be set to one if the DP master (Class 1) shall change the operation mode in case of an error. Otherwise this bit shall be set to zero.

### 5.11.8 Coding of the field Min_Slave_Interval

The coding of this field shall be as data type Unsigned16 with the value range from 1 to $2^{16}$-1 and the time base for the timer shall be 100 µs.

### 5.11.9 Coding of the field Poll_Timeout

The coding of this field shall be as data type Unsigned16 with the value range from 1 to $2^{16}$-1 and the time base for the timer shall be 1 ms.

### 5.11.10 Coding of the field Data_Control_Time

The coding of this field shall be as data type Unsigned16 with the value range from 1 to $2^{16}$-1 and the time base for the timer shall be 10 ms.

### 5.11.11 Coding of the field Alarm_Max

The coding of this field shall be as data type Unsigned8 with the value range from 7 to 32.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.11.12 Coding of the field Max_User_Global_Control

The coding of this field shall be as data type Unsigned8 with the value range from one to 255. The value Zero is reserved.

NOTE This parameter defines the maximum number of Global_Control requests, which may be started by the User at the same time. The parameter describes the ability of the DP-master. A practicable value for Max_User_Global_Control is 16. For each of the 8 Slave groups one SYNC and one FREEZE command may be started at the same time.

### 5.11.13 Coding of the field Master_User_Data_Len

This field shall contain the length of Master_User_Data inclusive the field Master_User_Data_Len itself. This field shall be coded as data type Unsigned16. The range of values shall be from 2 to ($2^{16}$-1).

### 5.11.14 Coding of the field Master_Class2_Name

The coding of this field shall be as data type Visible String with the fixed length of 32.

### 5.11.15 Coding of the field Master_User_Data

One of the following data types shall be used for each Master_User_Data field:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.11.16 Coding of the field $T_{CT}$

The coding of this field shall be as data type Unsigned32 with the value range from 1 to $2^{24}$ - 1.

### 5.11.17 Coding of the field $maxT_{SH}$

The coding of this field shall be as data type Unsigned8 with the value range from 1 to $2^{8}$-1.

## 5.12 Coding section related to the slave parameter set

### 5.12.1 Coding of the field Slave_Para_Len

The coding of this field shall be as data type Unsigned16 and the value for the Slave_Para_Len shall be in the range from zero to $2^{16}$ - 1.This parameter shall contain the length of Slave_Para inclusive the length parameter itself. A DP-slave parameter set can be deleted by setting the Slave_Para_Len to zero.

### 5.12.2 Coding of the field SI_Flag (slave flag)

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: reserved**

**Bit 1: Extra_Alarm_SAP**
This bit shall be set if the DP-master (Class 1) shall acknowledge alarms via SAP 50. Otherwise, if the DP-master (Class 1) acknowledges alarms via SAP 51 this bit shall not be set.

**Bit 2: DPV1_Data_Types**
This bit shall be set if extended data types are used in configuration (Extended_Special_Identifier_Format). Otherwise, if only octet or word types with the Identifier_Format or Special_Identifier_Format this bit may not be set.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 3: DPV1_Supported**
This bit shall be set to enable the DPV1 extensions. Otherwise the extensions shall be disabled.

**Bit 4: Publisher_Enable**
This bit shall be set to enable the publisher function. Otherwise the publisher function shall be disabled.

**Bit 5: Fail_Safe**
This bit shall be set to force the DP-master (Class 1) to convey a NULL-PDU during operation mode Clear. Otherwise data with the value zero will be sent during Clear.

**Bit 6: New_Prm**
This bit shall be set to force the DP-master (Class 1) to convey a new Set_Prm-REQ-PDU during the data transfer phase. Otherwise this bit shall be set to zero.

**Bit 7: Active**
This bit shall be set to force the DP-master (Class 1) to activate the corresponding DP-slave. Otherwise this bit shall be set to zero.

### 5.12.3 Coding of the field Slave_Type

This field shall be coded as data type Unsigned8 with values according to Table 25:

**Table 25 – Values for Slave_Type**

| Value (decimal) | Meaning |
|---|---|
| 0 | DP-slave |
| 1 to 15 | reserved |
| 16 to 255 | manufacturer specific |

### 5.12.4 Coding of the field Max_Diag_Data_Len

This field shall be coded as data type Unsigned8 and with values from the range 6 to 244.

### 5.12.5 Coding of the field Max_Alarm_Len

This field shall be coded as data type Unsigned8 and with values from the range 4 to Min(Max_Diag_Data_Len-6, 64).

### 5.12.6 Coding of the field Max_Channel_Data_Length

This field shall be coded as data type Unsigned8 and with values from the range 4 to 244.

NOTE This parameter defines the maximum APDU size for the corresponding DP-slaves on the MS1-AR.

### 5.12.7 Coding of the field Diag_Upd_Delay

This field shall be coded as data type Unsigned8 and with values from the range 0 to 15 (extendible up to 255).

NOTE The parameter is used to count the number of Slave_Diag.cnf in the state DIAG2 while Diag_Data.Prm_Req is still set (for Slaves with reduced performance).

### 5.12.8 Coding of the field Alarm_Mode

This parameter specifies the maximum number of possible active alarms.

This field shall be coded as data type Unsigned8 with values according to Table 26:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 26 – Values for Alarm_Mode**

| Value (decimal) | Meaning |
|---|---|
| 0 | 1 alarm of each type |
| 1 | 2 alarms in total |
| 2 | 4 alarms in total |
| 3 | 8 alarms in total |
| 4 | 12 alarms in total |
| 5 | 16 alarms in total |
| 6 | 24 alarms in total |
| 7 | 32 alarms in total |

### 5.12.9 Coding of the field Add_Sl_Flag

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: NA_To_Abort**
This bit shall be set if the DP-master (Class 1) shall proceed with data exchange in case the corresponding DP-slave is not responding. Otherwise this bit shall not be set.

**Bit 1: Ignore_AClr**
This bit shall be set if the DP-master (Class 1) shall exclude the corresponding DP-slave from the auto-clear behaviour. Otherwise this bit shall not be set.

**Bit 2 to 7: reserved**

### 5.12.10 Coding of the field MS1_Timeout

This field shall be coded as data type Unsigned16 with values from the range decimal 1 to $2^{16}$-1. The value 0 is reserved but may be used for back-up values.

### 5.12.11 Coding of the field Prm_Data_Len

This parameter contains the length of Prm_Data inclusive the length parameter.

This field shall be coded as data type Unsigned16 and with values from the range 9 to 246.

### 5.12.12 Coding of the field Prm_Data

This field shall be coded as a Set_Prm-REQ-PDU.

### 5.12.13 Coding of the field Cfg_Data_Len

This parameter contains the length of Cfg_Data inclusive the length parameter.

This field shall be coded as data type Unsigned16 and with values from the range 3 to 246.

### 5.12.14 Coding of the field Cfg_Data

This field shall be coded as a Chk_Cfg-REQ-PDU.

### 5.12.15 Coding of the field Add_Tab_Len

This parameter contains the length of Add_Tab inclusive the length parameter.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

This field shall be coded as data type Unsigned16 and with values from the range 2 to $2^{16}$-31.

### 5.12.16 Coding of the field Number_of_Entries

This field contains the number of entries in the address table.

This field shall be coded as data type Unsigned16 and with values from the range 1 to 488.

NOTE This parameter contains the number of entries in the address assignment table in the host to the addressed DP-slave. At maximum 488 entries per DP-slave are allowed (maximal 244 configuration strings doubled for inputs and outputs are possible).

### 5.12.17 Coding of the field Add_Tab_Entry_Header

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: IO_Selection**
This bit shall be set if the following addresses belong to the output part. Otherwise this bit shall not be set and the following addresses belong to the input part.

**Bit 1: Address_Format_Selection**
This bit shall be set to one if the following addresses are coded as Unsigned32. This bit shall be set to zero if the following addresses are coded as Unsigned16.

**Bit 2 to 7: reserved**

### 5.12.18 Coding of the field I/O_Data_Length

This field shall be coded as data type Unsigned8 with the values from one to 244.

### 5.12.19 Coding of the field I/O_Config_Address

This field represents the local address of the related configuration identifier.

This field shall be coded as data type Unsigned16 if the bit Address_Format_Selection in the field Add_Tab_Entry_Header is set to zero.

This field shall be coded as data type Unsigned32 if the bit Address_Format_Selection in the field Add_Tab_Entry_Header is set to one.

### 5.12.20 Coding of the field Host_Address

This field represents the local address of the related input or output data.

This field shall be coded as data type Unsigned16 if the bit Address_Format_Selection in the field Add_Tab_Entry_Header is set to zero.

This field shall be coded as data type Unsigned32 if the bit Address_Format_Selection in the field Add_Tab_Entry_Header is set to one.

### 5.12.21 Coding of the field Slave_User_Data_Len

This parameter contains the length of Slave_User_Data inclusive the length parameter.

This field shall be coded as data type Unsigned16 and with values from the range 2 to $2^{16}$-31.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.12.22 Coding of the field Slave_User_Data

One of the following data types shall be used for each Slave_User_Data field:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

#### 5.12.23 Coding of the field Ext_Prm_Data_Len

This parameter contains the length of Ext_Prm_Data inclusive the length parameter (2 octets).

This field shall be coded as data type Unsigned16 and with values from the range 2 (no Ext_Prm_Data follow) and 7 to 246.

#### 5.12.24 Coding of the field Ext_Prm_Data

This field shall be coded as a Set_Ext_Prm-REQ-PDU.

### 5.13 Coding section related to statistic counters

#### 5.13.1 Coding of the field DLPDU_sent_count and SD_count

The coding of this field shall be as data type Unsigned32.

#### 5.13.2 Coding of the field Error_count and SD_error_count

The coding of this field shall be as data type Unsigned16.

### 5.14 Coding section related to set slave address PDU

#### 5.14.1 Coding of the field New_Slave_Add

This field shall be coded as data type Unsigned8 with values from the range zero to 125.

#### 5.14.2 Coding of the field No_Add_Change

This field shall be coded as data type Boolean. The value TRUE shall be used to indicate that the change of the address is only once possible. The value FALSE shall be used to indicate that the change of the address is more than once possible.

#### 5.14.3 Coding of the field Rem_Slave_Data

One of the following data types shall be used for each Rem_Slave _Data field:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.15 Coding section related to initiate/abort PDUs

#### 5.15.1 Coding of the field Features_Supported_1

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: Read_Write**
This bit shall be set to indicate that Read and Write services on MS2-AR are supported.

NOTE Read and Write are mandatory on a MS2-AR.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 1 to 7: reserved**

### 5.15.2 Coding of the field Features_Supported_2

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

Bit 0 to 7: reserved

### 5.15.3 Coding of the field Profile_Features_Supported_1

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

Bit 0 to 7: defined by profile

This field shall be set to zero (decimal) if no profile is used. Otherwise the value shall be taken from the appropriate profile specification.

### 5.15.4 Coding of the field Profile_Features_Supported_2

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

Bit 0 to 7: defined by profile

This field shall be set to zero (decimal) if no profile is used. Otherwise the value shall be taken from the appropriate profile specification.

### 5.15.5 Coding of the field Profile_Ident_Number

The coding of this field shall be as data type Unsigned16. The value zero shall be set if no profile is used. Otherwise the value shall be taken from the appropriate profile specification.

### 5.15.6 Coding of the field S_Type (source type)

The coding of this field shall be as data type Unsigned8. The value zero shall be set to indicate the use of the short address format. The value one shall be set to indicate the use of the long address format. Other values are reserved.

NOTE The terms source and destination always belong to the direction of the used PDU.

### 5.15.7 Coding of the field D_Type (destination type)

The coding of this field shall be as data type Unsigned8. The value zero shall be set to indicate the use of the short address format. The value one shall be set to indicate the use of the long address format. Other values are reserved.

### 5.15.8 Coding of the field S_Len (source length)

This field shall be coded as data type Unsigned8.

### 5.15.9 Coding of the field D_Len (destination length)

This field shall be coded as data type Unsigned8.

### 5.15.10 Coding of the field S_API (source application identifier)

This field shall be coded as data type Unsigned8.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.15.11 Coding of the field D_API (destination application identifier)

This field shall be coded as data type Unsigned8.

#### 5.15.12 Coding of the field S_SCL (source security level)

This field shall be coded as data type Unsigned8. The value zero shall be set to indicate that no access level is used.

#### 5.15.13 Coding of the field D_SCL (destination security level)

This field shall be coded as data type Unsigned8. The value zero shall be set to indicate that no access level is used.

#### 5.15.14 Coding of the field S_Network_Address

The field shall only be present if the field S_Type is set to one.

The coding of this field shall be as data type Octet String with the fixed length of 6.

#### 5.15.15 Coding of the field D_Network_Address

The field shall only be present if the field D_Type is set to one.

The coding of this field shall be as data type Octet String with the fixed length of 6.

#### 5.15.16 Coding of the field S_MAC_Address

The field shall only be present if the field S_Type is set to one.

The coding of this field shall be as data type Octet String.

#### 5.15.17 Coding of the field D_MAC_Address

The field shall only be present if the field D_Type is set to one.

The coding of this field shall be as data type Octet String.

#### 5.15.18 Coding of the field Send_Timeout

This field shall be coded as data type Unsigned16. The time base for the timer shall be 10 ms. The values shall be set from the range 1 to $2^{16}$-1.

#### 5.15.19 Coding of the field Server_SAP

This field shall be coded as data type Unsigned8. The values shall be set from the range zero to 48.

#### 5.15.20 Coding of the field Subnet

The coding of this field shall be as data type Unsigned8 and the values shall be set according to Table 27.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 27 – Values for Subnet**

| Value (decimal) | Meaning |
|---|---|
| 0 | means no subnet |
| 1 | means subnet local |
| 2 | means subnet remote |
| 3 to 255 | reserved |

### 5.15.21 Coding of the field Instance_Reason_Code

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0 to 3: Reason Code**
The values shall be set according to Table 28 if Instance is set to DLL.

**Table 28 – Values of reason code if instance is DLL**

| Value (decimal) | Meaning |
|---|---|
| 1 | UE (meaning see Part 3) |
| 2 | RR (meaning see Part 3) |
| 3 | RS (meaning see Part 3) |
| 9 | NR (meaning see Part 3) |
| 10 | DH (meaning see Part 3) |
| 11 | LR (meaning see Part 3) |
| 12 | RDL (meaning see Part 3) |
| 13 | RDH (meaning see Part 3) |
| 14 | DS (meaning see Part 3) |
| 15 | NA (meaning see Part 3) |

The values shall be set according to Table 29 if Instance is set to MS2.

**Table 29 – Values of reason code if instance is MS2**

| Value (decimal) | Meaning |
|---|---|
| 1 | ABT_SE means sequence error |
| 2 | ABT_FE means invalid request APDU received |
| 3 | ABT_TO means connection timed out |
| 4 | ABT_RE means invalid response APDU received |
| 5 | ABT_IV means invalid service from the User |
| 6 | ABT_STO means requested value of Send_Timeout was to short |
| 7 | ABT_IA means invalid additional address information |
| 8 | ABT_OC means S-Timer expired, response APDU has not been sent yet |

**Bit 4 to 5: Instance**
The values shall be set as follows:

Decimal (0): DLL

Decimal (1): MS2

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

Decimal (2): USER

Decimal (3): reserved

**Bit 6 to 7: reserved**

### 5.16 Coding section related to read/write/data transport PDUs

#### 5.16.1 Coding of the field Index

This field shall be coded as data type Unsigned8 with values from the range zero to 255.

#### 5.16.2 Coding of the field Length

This field contains the number of octets of the Data field.

This field shall be coded as data type Unsigned8 with values from the range zero to 240.

### 5.17 Coding section related to load region and function invocation PDUs

#### 5.17.1 Coding of the field Extended_Function_Num

In addition to the field Function_Num, as described in 5.8.1, this field specifies the function of the Load Region or Function Invocation service.

The coding of this field shall be as data type Unsigned8 and the values shall be set according to Table 30:

**Table 30 – Values of Extended_Function_Num**

| Value (decimal) | Meaning |
|---|---|
| 0 | Initiate_Load |
| 1 | Push |
| 2 | Pull |
| 3 | Terminate |
| 4 | Start |
| 5 | Stop |
| 6 | Resume |
| 7 | Reset |
| 8 | Call |
| 9 | Get_FI_State |
| 10 to 255 | reserved |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.17.2 Coding of the field Options

The coding of this field shall be according to 3.4 and the individual bits shall have the following meaning:

**Bit 0: More_follows**
This bit is only valid for the Push APDU and Pull APDU. This bit shall be set if the Load Region sequence is not finished and additional Push APDUs or Pull APDUS follow subsequently. This bit shall be reset, if the sequence is finished and no additional Push APDUs or Pull APDUS follow.

For the Initiate_Load APDU this bit shall be reset.

**Bit 1 to 6: reserved**

**Bit 7: Pull / Push**
This bit is only valid in the Initiate_Load APDU. This bit shall be set, if Pull LR sequence is initialised. This bit shall be reset, if a Push LR sequence is initialised.

For the Push APDU and Pull APDU this bit shall be reset.

#### 5.17.3 Coding of the field Sequence_Number

This field shall be coded as data type Unsigned32.

#### 5.17.4 Coding of the field LR_Data

This field shall be coded as data type Octet String.

#### 5.17.5 Coding of the field Max_Segment_Length

This field shall be coded as data type Unsigned8.

#### 5.17.6 Coding of the field LR_Index

This field shall be coded as data type Unsigned16.

#### 5.17.7 Coding of the field LR_Length

This field shall be coded as data type Unsigned32.

#### 5.17.8 Coding of the field Max_Response_Delay

This field shall be coded as data type Unsigned16.

#### 5.17.9 Coding of the field Intersegment_Request_Timeout

This field shall be coded as data type Unsigned16.

#### 5.17.10 Coding of the field User_Specific

This field contains userspecific information, e.g. the format of the LR_Data.

This field shall be coded as data type Octet String.

#### 5.17.11 Coding of the field FI_Index

This field shall be coded as data type Unsigned16.

Allowed values: The available index range is structured as follows:
- Indices reserved for manufacturer specific use (0 - 32767)
- Indices reserved for Profibus system services (32768 - 64999)
- Indices reserved for Identification and Maintenance (I&M) services (65000 - 65199).

#### 5.17.12 Coding of the field Entity Number

This field shall be coded as data type Unsigned8.

Allowed values: The meaning of this parameter is profile specific. If not used it shall be set to 0.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.17.13 Coding of the field Execution_Argument

This field shall be coded as data type Octet String.

### 5.17.14 Coding of the field Result_Argument

This field shall be coded as data type Octet String.

### 5.17.15 Coding of the field FI_State

This field contains the state of the Function Invocation entity.

The coding of this field shall be as data type Unsigned8 and the values shall be set according to Table 31:

**Table 31 – Values of FI_State**

| Value (decimal) | Meaning |
|---|---|
| 0 | IDLE |
| 1 | STARTING |
| 2 | RUNNING |
| 3 | STOPPING |
| 4 | STOPPED |
| 5 | RESUMING |
| 6 | UNRUNNABLE |
| 7 | RESETTING |
| 8 | WAIT-SET-IN-USE |
| 9 | WAIT-SET-IN-USE-2 |
| 10 | WAIT-DEL-IN-USE |
| 11 | WAIT-DEL-IN-USE-2 |
| 12 | EXEC-ERR |
| 13 | EXEC-TERM |
| 14 to 255 | reserved |

This field shall be coded as data type OctetString with 10 octect. The value shall be set manufacturer specific and shall be filled with 0x00 if it is shorter than 16.

### 5.17.16 Coding of the field IMData_Execution_Argument

#### 5.17.16.1 General

This field contains the IMData_Execution_Argument of the CALL-REQ-PDU. The coding of the field is specified for:

- IM_Date in IEC 61158-6-10, 6.2.4.83.15,
- IM_Descriptor in IEC 61158-6-10, 6.2.4.83.16,
- IM_Header in 5.17.16.2,
- I&M_Manufacturer_Specific_x in 5.17.16.5,
- IM_Profile_Specific_Data in 5.17.16.4,
- IM_Signature in IEC 61158-6-10, 6.2.4.83.17,
- IM_Tag_Function in IEC 61158-6-10, 6.2.4.83.13, and
- IM_Tag_Location in IEC 61158-6-10, 6.2.4.83.14.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

#### 5.17.16.2 Coding of the field IM_Header

This field shall be coded as data type OctetString with 10 octect. The value shall be set manufacturer specific and shall be filled with 0x00 if it is shorter than 16.

#### 5.17.16.3 Coding of the field IM_Manufacturer_ID

This field shall be coded as data type Unsigned16.

#### 5.17.16.4 Coding of the field IM_Profile_Specific_Data

This field shall be coded as data type OctetString with 54 octets. The value shall be set profile specific and shall be filled with 0x00 if it is shorter than 54.

#### 5.17.16.5 Coding of the field I&M_Manufacturer_Specific_x

This field shall be coded as data type OctetString with 64 octets. The value shall be set manufacturer specific and shall be filled with 0x00 if it is shorter than 64.

### 5.17.17 Coding of the field IMData_Result_Argument

This field contains the IMData_Result_Argument of the CALL-RES-PDU. The coding of the field is specified for:

- IM_Date in IEC 61158-6-10, 6.2.4.83.15,
- IM_Descriptor in IEC 61158-6-10, 6.2.4.83.16,
- IM_Hardware_Revision in IEC 61158-6-10, 6.2.4.83.3,
- IM_Header in 5.17.16.2,
- IM_Manufacturer_ID in 5.17.16.3,
- I&M_Manufacturer_Specific_x in 5.17.16.5,
- IM_Profile_ID in IEC 61158-6-10, 6.2.4.83.8,
- IM_Profile_Specific_Data in 5.17.16.4,
- IM_Profile_Specific_Type in IEC 61158-6-10, 6.2.4.83.9,
- IM_Revision_Counter in IEC 61158-6-10, 6.2.4.83.7,
- IM_Serial_Number 6.2.4.83.2,
- IM_Signature in IEC 61158-6-10, 6.2.4.83.17,
- IM_Supported in IEC 61158-6-10, 6.2.4.83.12,
- IM_SWRevision_Bug_Fix in IEC 61158-6-10, 6.2.4.83.5,
- IM_SWRevision_Functional_Enhancement in IEC 61158-6-10, 6.2.4.83.4,
- IM_SWRevision_Internal_Change in IEC 61158-6-10, 6.2.4.83.6,
- IM_Tag_Function in IEC 61158-6-10, 6.2.4.83.13,
- IM_Tag_Location in IEC 61158-6-10, 6.2.4.83.14,
- IM_Version_Major in IEC 61158-6-10, 6.2.4.83.10,
- IM_Version_Minor in IEC 61158-6-10, 6.2.4.83.11,
- OrderID in IEC 61158-6-10, 4.8.2.2.2, and
- SWRevisionPrefix in IEC 61158-6-10, 4.8.2.2.4.

## 5.18 Examples of diagnosis-RES-PDUs

Figure 8 illustrates an example for a diagnosis APDU with a device related status message, a device related process alarm and an identifier related diagnostic.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | Headeroctet: device related diagnostic |
| 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Status_Type: Status_Message |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Slot_Number: 2 (sensor A) |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Specifier: no further differentiation |
| 5 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Diagnostic_User_Data: 5 (average temperature) |
| 6 | | | | | | | | | Diagnostic_User_Data: temperature |
| 7 | | | | | | | | | value (Unsigned16) |
| 8 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | Headeroctet: device related diagnostic |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Alarm_Type: Process_Alarm |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Slot_Number: 3 (valve B) |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Specifier: alarm appears |
| 12 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | Diagnostic_User_Data: 0x50 (upper limit exceeded pressure) |
| 13 | | | | | | | | | Diagnostic_User_Data: time stamp |
| 14 | | | | | | | | | (Time_of_day = 4 octets) |
| 15 | | | | | | | | | |
| 16 | | | | | | | | | |
| 17 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | Headeroctet: identifier related diagnostic |
| 18 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1st identifier Number with diagnosis |
| | msb | | | | | | | | |

**Figure 8 – Example of Ext_Diag_Data in case of DPV1 diagnosis format with alarm and status PDU**

**Corresponding textual part**
;text assignments for sensor A and valve B

Unit_Diag_Area = 24-27

Value(1) = “Minimum temperature”

Value(2) = “Maximum temperature”

Value(5) = “Average temperature”

Unit_Diag_Area_End

Unit_Diag_Area = 28-31

Value(1) = “lower limit exceeded pressure”

Value(5) = “upper limit exceeded pressure”

Unit_Diag_Area_End

Unit_Diag_Area = 8-15

Value(2) = “sensor A”

Value(3) = “valve B”

Unit_Diag_Area_End

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

Unit_Diag_Area = 16-17

Value(1) = “alarm/status appearing”

Value(2) = “alarm/status disappearing”

Unit_Diag_Area_End

Since these definitions are used for both alarms and status messages their values should be different. That means different values for alarms and status messages should be used at the same position within the diagnostic field.

Figure 9 illustrates an example for a diagnosis APDU with a device related userspecific diagnostic, an identifier related diagnostic and a channel related diagnostic.

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets<br>1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | device related diagnostic: |
| 2 | device specific | | | | | | | | meaning of the bits is defined |
| 3 | diagnostic field | | | | | | | | manufacturer specific |
| 4 | of length 3 | | | | | | | | |
| 5 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | identifier related diagnostic: |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | identifier number 0 has diagnostic |
| 7 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | identifier number 12 has diagnostic |
| 8 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | identifier number 18 has diagnostic |
| 9 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | channel related diagnostic:<br>identifier number 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | channel 2 |
| 11 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | overload, channel bit organised |
| 12 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | identifier number 12 |
| 13 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | channel 6 |
| 14 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | upper limit value exceeded, channel word organised |
| | msb | | | | | | | | |

**Figure 9 – Example of Ext_Diag_Data in case of the basic diagnosis format**

### 5.19 Example of Chk_Cfg-REQ-PDU

Figure 10 illustrates an example for a Chk_Cfg APDU with a special identifier format.

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets<br>1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | input/output, 3 octets manufacturer specific data |
| 2 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | consistency, output, 16 words |
| 3 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | consistency, input, 8 words |
| 4 | manufacturer | | | | | | | | |
| 5 | specific | | | | | | | | |
| 6 | data | | | | | | | | |
| | msb | | | | | | | | |

**Figure 10 – Example of a special identifier format**

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

### 5.20 Examples of Chk_Cfg-REQ-PDUs with DPV1 data types

The data types are coded into the manufacturer-specific part.

Simple data types are coded in a octet which contains the code of the data type.

Sequences of data types are coded as a list of simple data types. The sum of the lengths of the data types has to correspond to the input- or output data length. Data types with variable length (visible string, octet string, time of day, time difference) cannot be handled in a sequence, but as single element. Comments may be added at the end of the list of the data types.

Figure 11 shows the special identifier format for an analogue input block containing a value (Floating Point), Status (Unsigned8) and Comment (3 Octets).

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | Manufacturer-specific format |
| 2 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Consistency, 5 Octets (Input) |
| 3 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Data type Floating Point |
| 4 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Data type Unsigned8 |
| 5 | Manufacturer | | | | | | | | Comment |
| 6 | Specific | | | | | | | | Comment |
| 7 | Data | | | | | | | | Comment |
| | msb | | | | | | | | |

**Figure 11 – Example of a special identifier format with data types**

Figure 12 shows the special identifier format for an analogue output block where the output value can be read back.

The representation of this analogue output block is structured as follows:

Output Value (Floating Point)
Output Status (Unsigned8)
read back Value (Floating Point)
read back Status (Unsigned8)
comment (1 octet)

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | Manufacturer-specific format |
| 2 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Consistency, 5 Octets (Output) |
| 3 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Consistency, 5 Octets (Input) |
| 4 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Data Type Floating Point |
| 5 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Data Type Unsigned8 |
| 6 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Data Type Floating Point |
| 7 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Data Type Unsigned8 |
| 8 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Comment |
| | msb | | | | | | | | |

**Figure 12 – Example of a special identifier format with data types**

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Figure 13 shows the special identifier format for an empty slot with 3 octets comment.

The representation of this empty slot is structured as follows:

empty slot

comment (3 octets)

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets<br>1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Manufacturer-specific format |
| 2 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Comment |
| 3 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | Comment |
| 4 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | Comment |
| | msb | | | | | | | | |

**Figure 13 – Example of a empty slot with data types**

Figure 14 shows the use of the special identifier format for a multi-variable device which utilises the user specific coding instead of data types in the data type field.

The representation of this multi variable device is structured as follows:

Analog Input (AI)

Discrete Output (DO) with setpoint (SP_D) and readback (READBACK_D)

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Multi variable device, AI and DO |
|---|---|---|---|---|---|---|---|---|---|
| octets<br>1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | **Analog Input**<br>Special Config Identifier for the example AI (0x42) |
| 2 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Extended Length Octet for input (0x84) |
| 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | User specific instead of data type >128 ( e.g block code for AI 0x81) |
| 4 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | User specific (e.g. Identification of cyclic parameter for AI 0x81) |
| 5 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | **Discrete Output**<br>Special Config Identifier for the example DO (0xC2) |
| 6 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Extended Length Octet for output (0x81) |
| 7 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Extended Length Octet for input (0x81) |
| 8 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | User specific instead of data type >128 (e.g. block code for DO 0x84) |
| 9 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | User specific instead of data type >128 (e.g. Identification of cyclic parameter for DO with SP_D and READBACK_D 0x83) |
| | msb | | | | | | | | |

**Figure 14 – Example for multi-variable device with AI and DO function blocks**

### 5.21 Example structure of the Data_Unit for Data_Exchange

The structure of the Data_Unit for the service Data_Exchange is defined by the identifiers which are passed to the DP-slave with the service Chk_Cfg at configuration.

The following example illustrates how the data in the Data_Unit will be transferred, corresponding to the given identifiers. Figure 15 shows the data which shall be sent by Data_Exchange and their order as defined by the service Chk_Cfg.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| 1. ID | 2. ID | 3. ID | 4. ID | 5. ID | 6. ID | 7. ID | |
|---|---|---|---|---|---|---|---|
| 1 B_I | 1 W_I | 2 W_O | 2 B_IO | 1 B_O | 2 W_I | 3 W_A | |

**Figure 15 – Identifiers (ID)**

Figure 16 shows the type and length of the data corresponding to their Identifiers and the direction of their transfer, meaning wether they are Input or Output.

1 B_I: 1 Octet input;
1 B_O: 1 Octet output;
2 B_IO: 2 Octet out- and input
1 W_I: 1 Word input
1 W_O: 1 Word output
etc.

**Figure 16 – Identifier list**

The sequence of data in the Data_Unit is shown in Figure 17.

**Data_Exchange.req**

**msb**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | Word_high | | | | | | | | (3. ID) |
| 2 | Word_low | | | | | | | | (3. ID) |
| 3 | Data | | | | | | | | (4. ID) |
| 4 | Data | | | | | | | | (4. ID) |
| 5 | Data | | | | | | | | (5. ID) |
| 6 | 1.Word_high | | | | | | | | (7. ID) |
| 7 | 1.Word_low | | | | | | | | (7. ID) |
| 8 | 2.Word_high | | | | | | | | (7. ID) |
| 9 | 2.Word_low | | | | | | | | (7. ID) |
| 10 | 3.Word_high | | | | | | | | (7. ID) |
| 11 | 3.Word_low | | | | | | | | (7. ID) |
| 12 | etc. | | | | | | | | |
| … | | | | | | | | | |
| n | | | | | | | | | |

**Data_Exchange.res**

**msb**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | Data | | | | | | | | (1. ID) |
| 2 | Word_high | | | | | | | | (2. ID) |
| 3 | Word_low | | | | | | | | (2. ID) |
| 4 | Data | | | | | | | | (4. ID) |
| 5 | Data | | | | | | | | (4. ID) |
| 6 | 1.Word_high | | | | | | | | (6. ID) |
| 7 | 1.Word_low | | | | | | | | (6. ID) |
| 8 | 2.Word_high | | | | | | | | (6. ID) |
| 9 | 2.Word_low | | | | | | | | (6. ID) |
| 10 | etc. | | | | | | | | |
| 11 | | | | | | | | | |
| 12 | | | | | | | | | |
| … | | | | | | | | | |
| n | | | | | | | | | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Figure 17 – Structure of the Data_Unit for the request- and response-DLPDU**

For the configuration identifier “empty slot” no data will be transferred in the Data_Unit.

# 6 FAL protocol state machines

## 6.1 Overall structure

### 6.1.1 Fieldbus Service Protocol Machines (FSPM)

The FSPM State Machines co-ordinate the underlying state machines used for processing of the various services and application relations.

There exists one state machine for each device type (FSPMS for DP-slave, FSPMM1 for DP-master (Class 1) and FSPMM2 for DP-master (Class 2).

### 6.1.2 Master to Slave cyclic (MS0)

The MSCY1 State Machines are responsible for the cyclic data transfer between the DP-master (Class 1) and one DP-slave (MSCY1M/MSCY1S). This comprises the parameterisation, configuration, diagnostic and the data exchange as well as the Global Control commands. There exist one MSCY1S for a DP-slave and up to 125 MSCY1M for a DP-master (Class 1). To Subscribe Input Data objects a DP-slave use up to 125 SSCY1S.

### 6.1.3 Master (class 1) to Slave acyclic (MS1)

The MSAC1 State Machines (MSAC1M, MSAC1S) are responsible for the acyclic data transfer between the DP-master (Class 1) and one DP-slave. The MS1 communication relationship is coupled to the cyclic communication relationship and processes the services Read, Write and Alarm Ack as well as the Load Region and Function Invocation services.

Additionally the MSAL1M state machine is used for alarm handling at the DP-master.

There exist one MSAC1S for a DP-slave and up to 125 MSAC1M/MSAL1M for a DP-master (Class 1).

### 6.1.4 Master (class 2) to Slave acyclic (MS2)

The MS2 State Machines (MSAC2M, MSRM2S, MSAC2S) are responsible for the acyclic data transfer between the DP-master (Class 2) and one DP-slave. The MS2 communication relationship processes the services Read, Write and Data_Transport as well as the Load Region and Function Invocation services.

The Master Slave Resource Manager (MSRM2S) is responsible for the assignment of a DLSAP and the related resources to each established MSAC2 communication relationship in the DP-slave. The Master Slave Resource Manager additionally starts the appropriate MSAC2S State Machine.

At the DP-master (Class 2) an arbitrary number of MSAC2M state machines exist. The maximum number of MSAC2M state machines is given by the maximum number of DP-slaves (=125) where each DP-slave provides the maximum number of MS2 CRs (=49).

### 6.1.5 Master to Slave clock synchronisation (MS3)

By means of the MSCS1 State Machines the clock synchronisation can be accomplished between the DP-master (Class 1) and the DP-slaves.

There exist at most one state machine at DP-master (Class 1) and one at DP-slave.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

### 6.1.6 Master Master acyclic (MM1/MM2)

By means of the MMAC State Machines the Master_Diag, the Master Parameter Set, Slave Parameter Set, Bus Parameter Set can be transferred between the DP-master (Class 1) and the DP-master (Class 2).

There exist at most one state machine at DP-master (Class 1) and one at DP-master (Class 2).

### 6.1.7 DLL Mapping Protocol Machines (DMPM)

The DL Mapping Protocol Machines (DMPM) connects the other State machines and Layer 2. DMPM provides the coordination of all state machines concerning the configuration and error handling of the Data Link Layer Usage. The functions are mapped by the DMPM to the DLL and DLM services of Layer 2. The DMPM generates the necessary Layer 2 parameters of the service, receives the confirmations and indications from Layer 2 and passes them to the appropriate DMPM-User.

There exist one state machine for each device type (FSPMS for DP-slave, FSPMM1 for DP-master (Class 1) and FSPMM2 for DP-master (Class 2).

Interface between DMPM and Layer 2

The DMPM is directly put upon the DLL User – DLL interface and on the DLMS User – DLM interface as described in Data Link Layer Service Definitions (Refer to DL Service Definition in IEC 61158-3-3).

The DMPM uses for the mapping of the DMPM functions on the Layer 2 the following services:

a) DLL services
   - Send and Request Data with Reply (SRD)
   - Send Data with No Acknowledge (SDN)
   - Send and Request Data with Multicast Reply (MSRD)
   - Clock Synchronization (CS)

b) DLM services
   - Reset
   - Set Value
   - Get Value
   - Event
   - SAP Activate
   - SAP Activate Responder
   - SAP Deactivate

## 6.2 Assignment of state machines to devices

Table 32 shows the assignment of State Machines to devices. Some State Machines are always required and thus marked with ‘M’ (mandatory) others are marked with ‘O’ (optional). The Remarks show options in mandatory machines.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

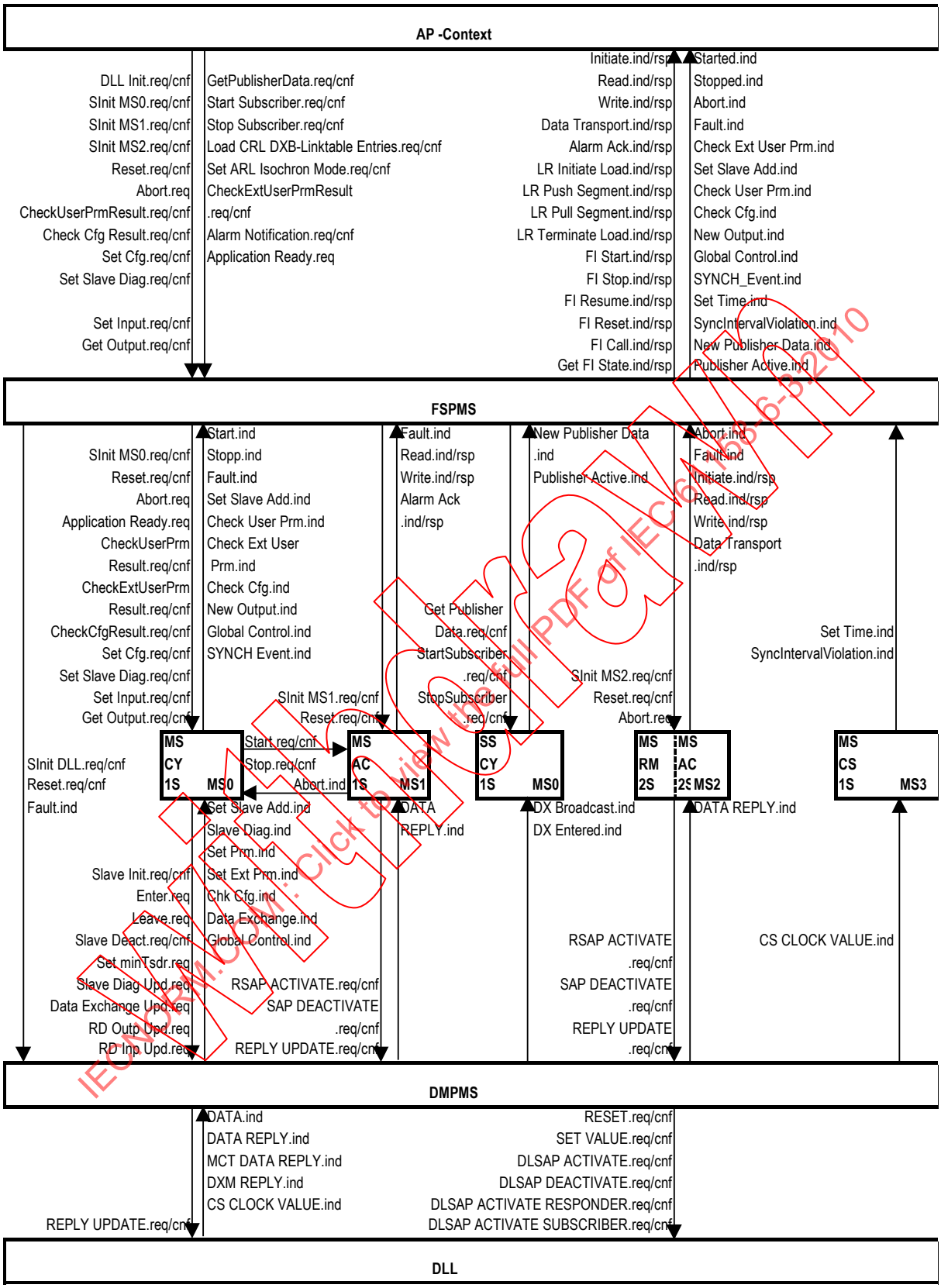
**Table 32 – Assignment of state machines**

| Device type | Machine | Mandatory/ optional | Remarks |
|---|---|---|---|
| DP-slave | FSPMS | M | Alarm handling is optional |
| | MSCY1S | M | |
| | SSCY1S | O | |
| | MSAC1S | O | |
| | MSAC2S | O | |
| | MSRM2S | O | |
| | DMPMS | M | |
| DP-master (Class 1) | FSPMM1 | M | |
| | MSCY1M | M | |
| | MSAL1M | O | |
| | MSAC1M | O | |
| | MMAC1 | O | |
| | DMPMM1 | M | |
| DP-master (Class 2) | FSPMM2 | M | |
| | MSAC2M | O | |
| | MMAC2 | O | |
| | DMPMM2 | M | |

## 6.3 Overview DP-slave

Figure 18 illustrates the general structure of the AL of a DP-slave by showing its state machines and the services used by them.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

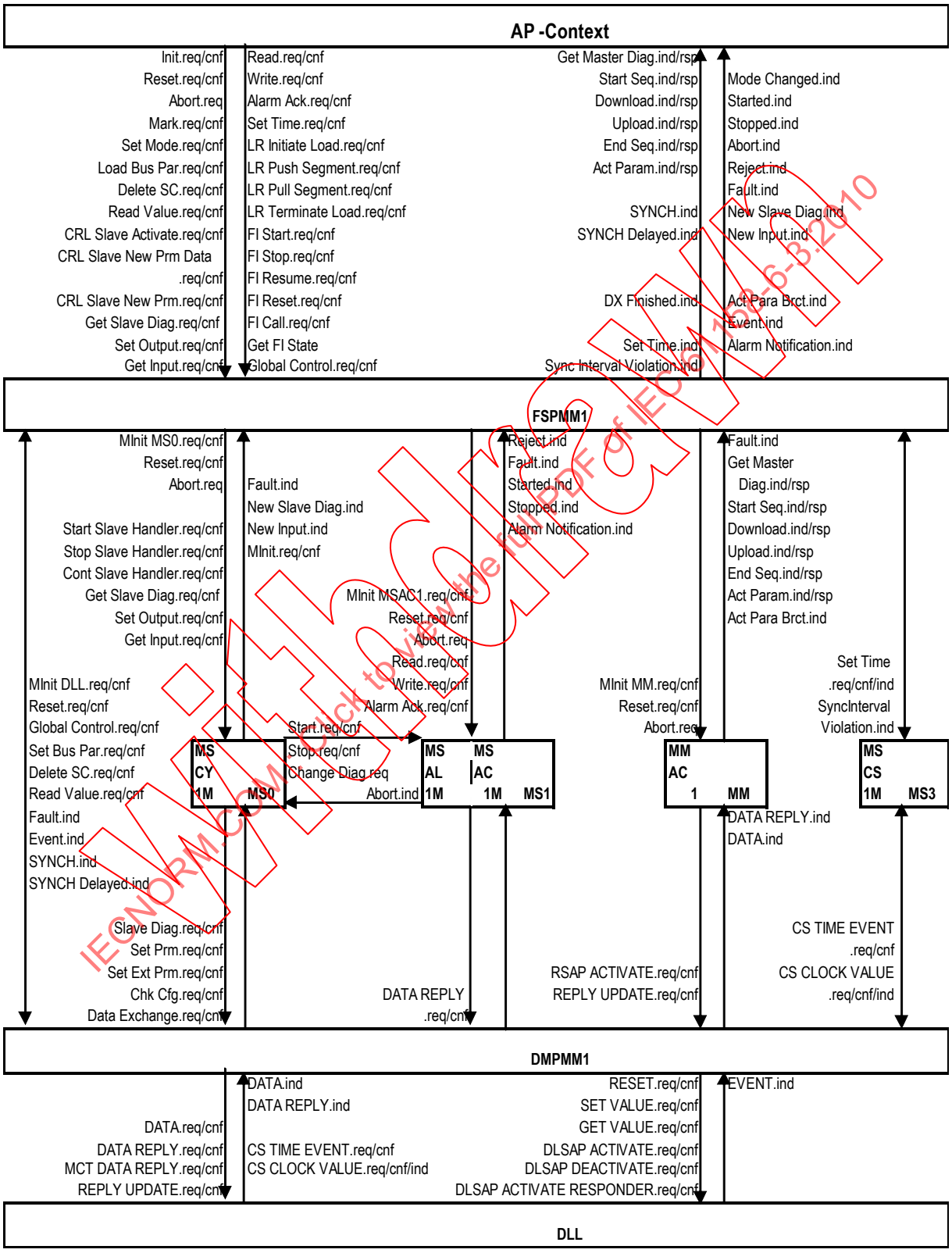



IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Figure 18 – Structuring of the protocol machines and adjacent layers in a DP-slave**

### 6.4 Overview DP-master (class 1)

Figure 19 illustrates the general structure of the AL of a DP-master (Class 1) by showing its state machines and the services used by them.

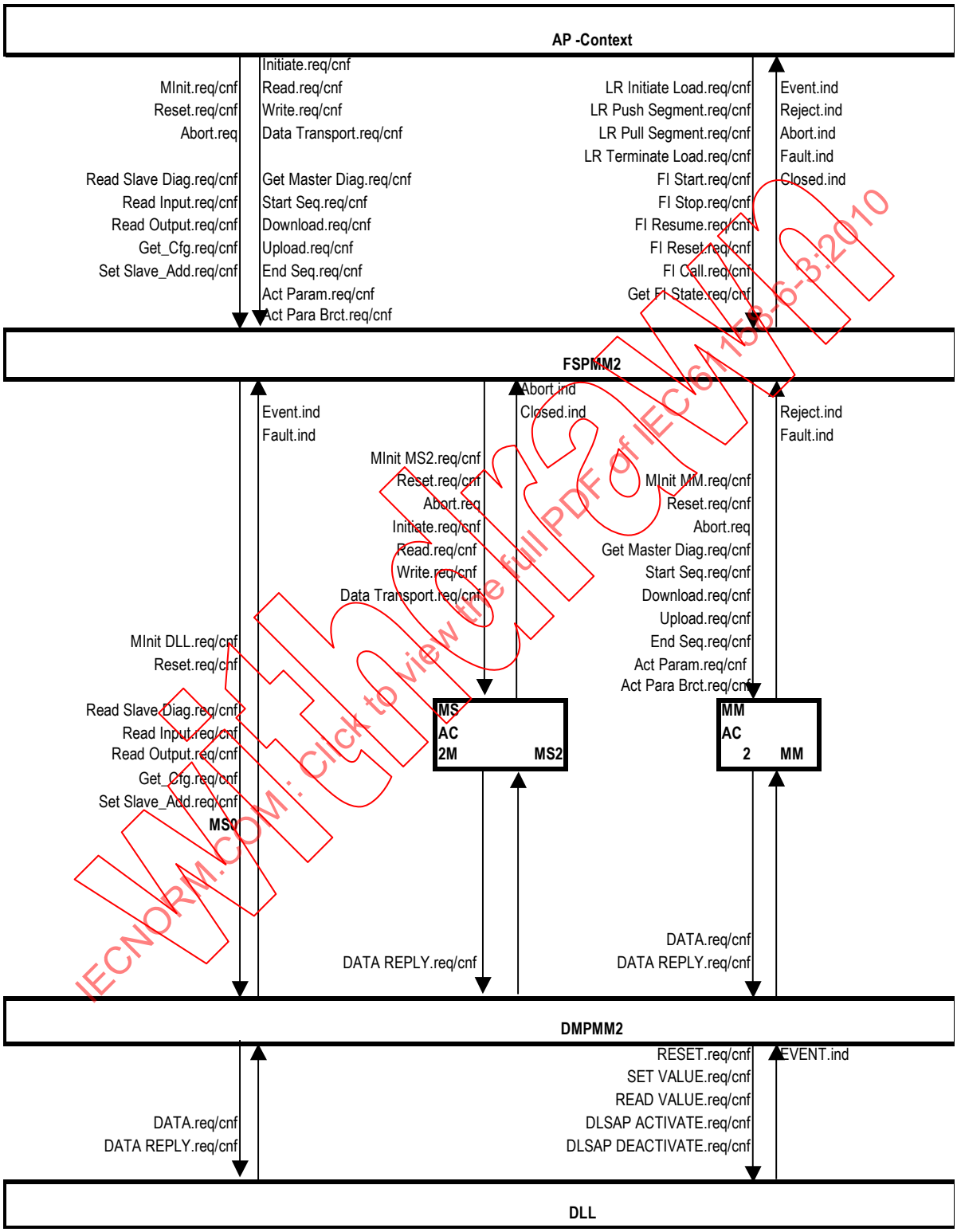



IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Figure 19 – Structuring of the protocol machines and adjacent layers in a DP-master (class 1)**

### 6.5 Overview DP-master (class 2)

Figure 20 illustrates the general structure of the AL of a DP-master (Class 2) showing its state machines and the services used.




IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Figure 20 – Structuring of the protocol machines and adjacent layers in a DP-master (class 2)**

### 6.6 Cyclic communication between DP-master (class 1) and DP-slave

The communication between DP-master and DP-slave is established with the sequence as shown in Figure 21:

If the DP-master wants to communicate with a DP-slave, the DP-master requests the Diag_Data of the DP-slave, in order to check the readiness for operation of the DP-slave.

This Diagnosis request will be repeated until the DP-slave responds with the requested data.

If the DP-slave responds with the requested diagnostic information, the DP-master checks whether another DP-master occupies this DP-slave. If this is not the case, the DP-slave will be parameterised and configured (Set_Prm / Chk_Cfg).

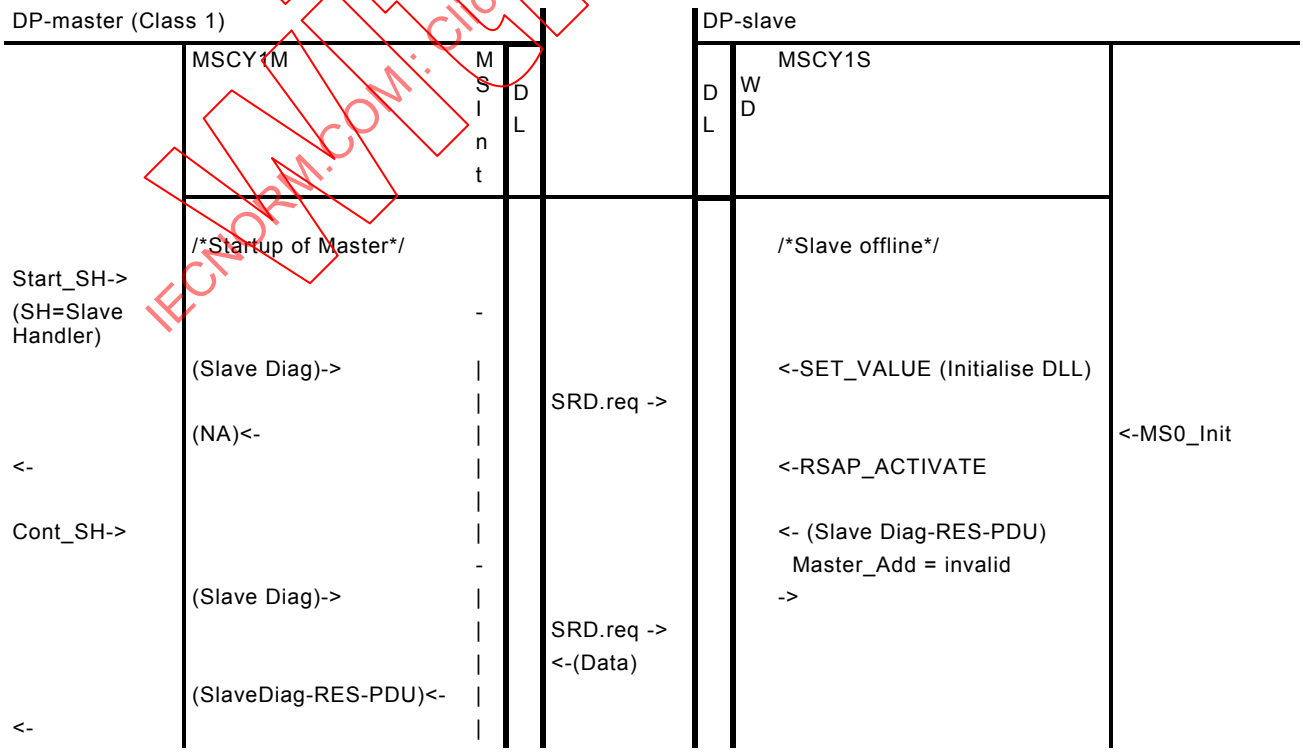
Subsequently the DP-master requests again the diagnostic data from the DP-slave. The following messages can occur:

- parameterisation or configuration error
- The DP-slave is occupied by another DP-master
- Static User diagnostics, no normal data transfer possible (for Example: start-up procedure for DP correctly executed, but some external parameterisation outstanding)
- DP-slave is not ready for operation

In the cases a) and b) the DP-master returns to the starting state and checks again the readiness for operation of the DP-slave.

In the cases c) and d) the DP-master requests the diagnostic information from the DP-slave until the mentioned messages disappear.

If no error messages occur the DP-master starts the data exchange to the DP-slave. The input and output data will be transferred to and from the DP-slave. The DP-master (Class 1) indicates his own operation mode to the DP-slaves for the first time.




IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

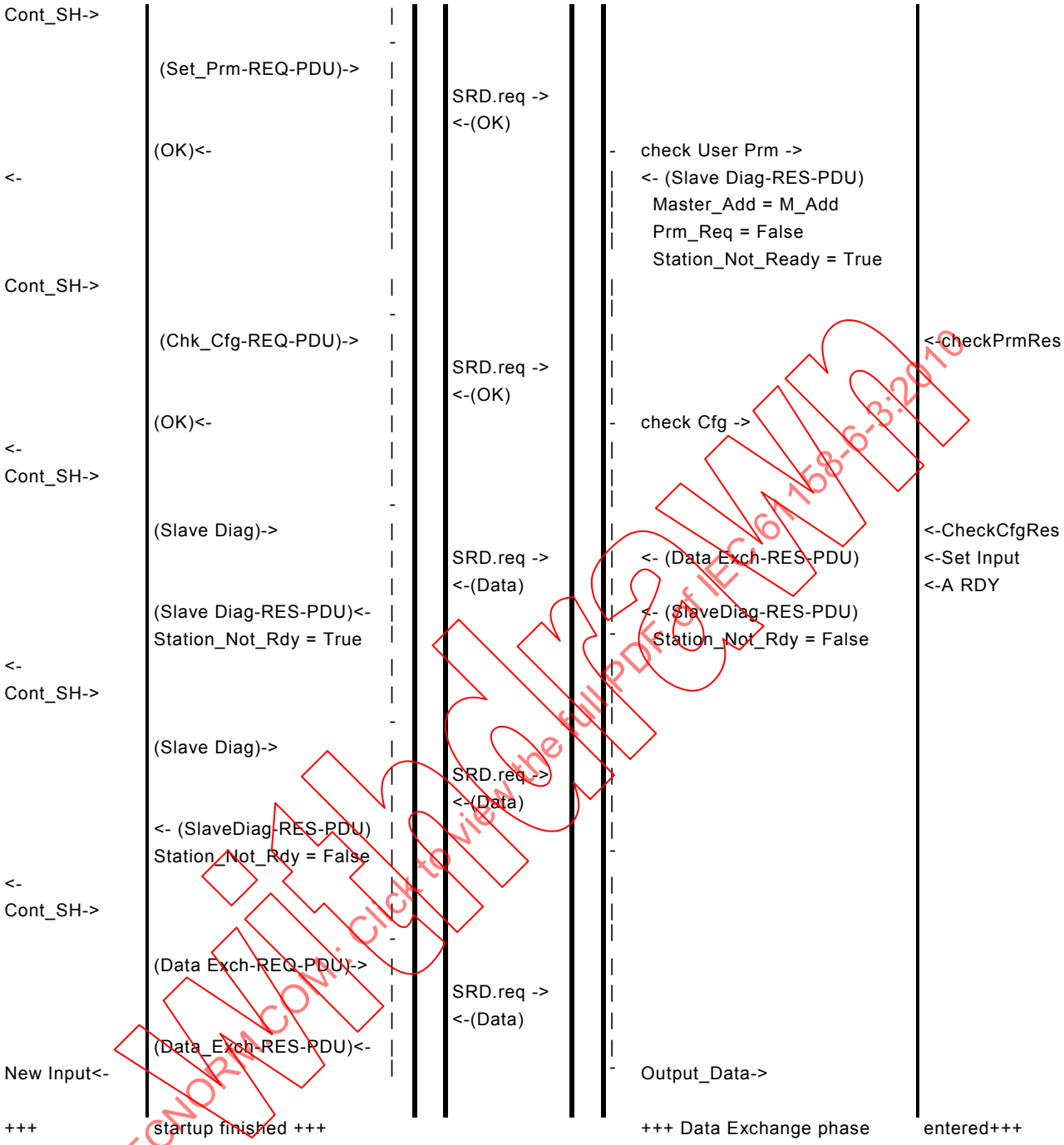



**Figure 21 – Sequence of the communication between DP-master and DP-slave**

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 6.7 Acyclic communication between DP-master (class 2) and DP-master (class 1)

The communication between DP-master (Class 2) and DP-master (Class 1) is handled in a fixed way (Request-Poll-Response), see Figure 22, with one exception.

This exception is the function Act_Para_Brct, which is transferred as a multicast from the DP-master (Class 2) to the corresponding DP-master (Class 1).

Communication is always initiated by the DP-master (Class 2). The DP-master (Class 2) sends a request to the DP-master (Class 1) and waits for the response. The service is processed by the MMAC2 of the DP-master (Class 2). After the receipt of the response the DP-master (Class 2) can make a new request.

At one point of time the DP-master (Class 1) can only communicate with one DP-master (Class 2).

The DP-master communication is initiated by a xxx.req from the DP-master (Class 2). The MMAC1 passes this request to Layer 2 and checks the status after receipt of the DMPM-response. If the response status is NR (no reply data) it will continue with a new SRD with no L_sdu to get response data. This procedure will be repeated by the DP-master (Class 2) until

an erroneous response occurs

or a SRD_RES_PDU with response data is received

or the Timer T1 expires.

The result will then be passed to the User of the DP-master (Class 2).

The timer T1 will be started after receipt of the xxx.req in the MMAC2 of the DP-master (Class 2) and will be stopped after release of the xxx.cnf. The value loaded in the timer T1 depends on the DLL parameter and the performance of the DP-master (Class 1).

The responder (DP-master (Class 1)) waits after a reset for a request. If the MMAC1 of the DP-master (Class 1) receives a valid request, this request is passed to the User by the corresponding xxx.ind. At this point of time an access protection was set to the local service access point (SAP) for the communication with the DP-master (Class 2). This is necessary to make sure that the response data will be sent to the right DP-master (Class 2).

After receipt of the response from the User (xxx.rsp) the response data will be passed with a REPLY_UPDATE.req to DMPM. Additionally, a timer will be started which controls the request for the response. If this timer expires and the response data are still not fetched by the DP-master (Class 2), the MMAC1 deletes the response data because the MMAC1 presumes that the DP-master (Class 2) is no longer available. In this case the access protection for the DP-master (Class 2) will be cancelled. This will also be done in the case of a single transaction if a DATA_REPLY.ind indicates that the response data are fetched by the DP-master (Class 2). The value loaded in the timer of the DP-master (Class 1) depends on the DL parameter and the performance of the DP-master (Class 2).

In case of Master-Master communication a sequence of requests can put into parentheses. Therefore the access protection will be set by the “Start_Seq” and released by the “End_Seq”. To avoid errors the time between two requests will be supervised. The value for the time supervision is added to the Start_Seq request. With the Start_Seq.req the User can additionally send an areacode. By means of this areacode at the beginning of a sequence an area can be reserved at the User in the DP-master (Class 1).

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

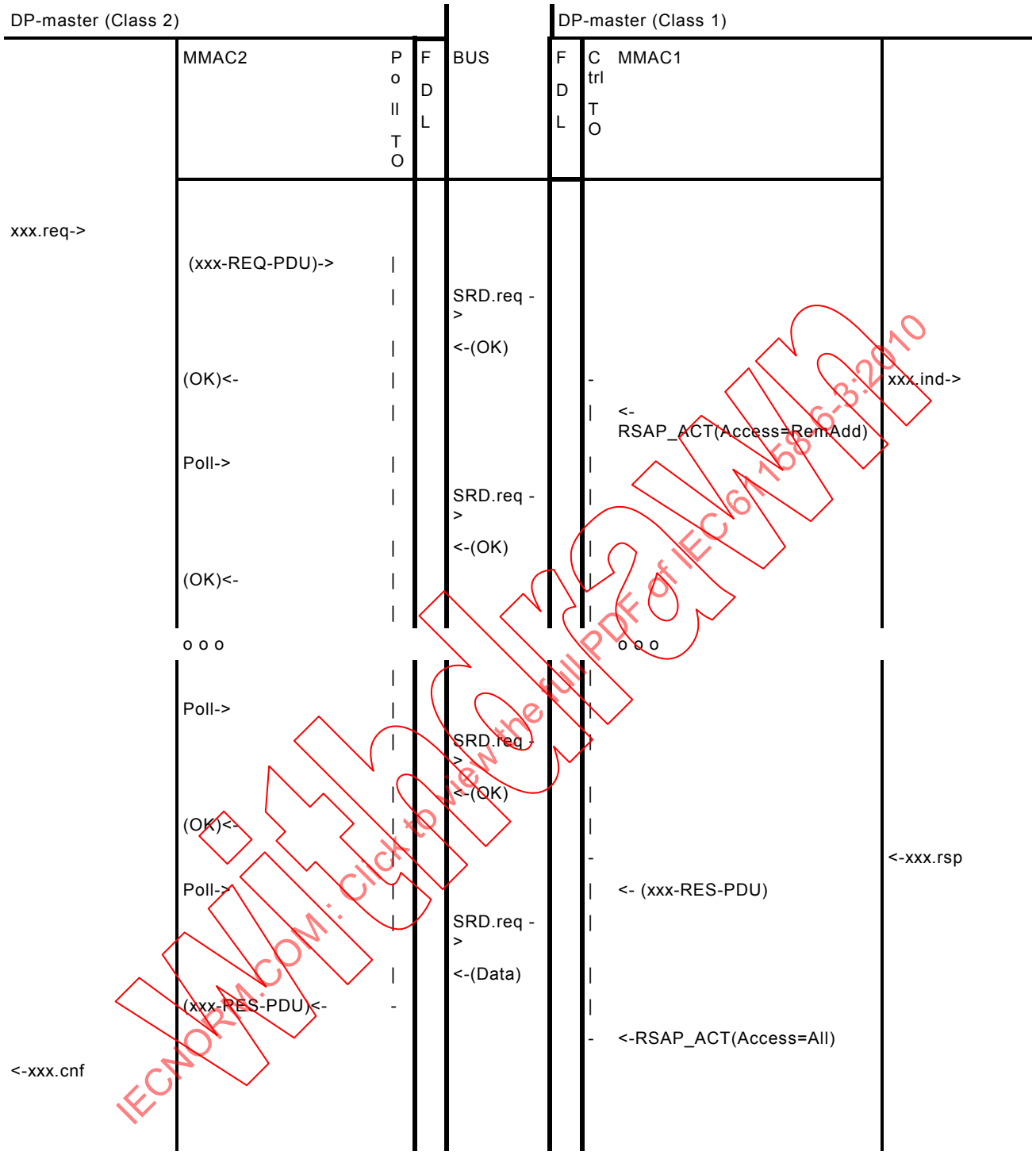



**Figure 22 – Sequence of communication between DP-master (class 2) and DP-master (class 1)**

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 6.8 Acyclic communication between DP-master (class 1) and DP-slave

The acyclic communication between DP-master (Class 1) and DP-slave is handled in a fixed way (Request-Poll-Response) on the MS1 application relationship, see Figure 23.

The communication is always initiated by the DP-master (Class 1). The connection is established if the cyclic communication MS0 AR has entered the data exchange mode. The DP-master (Class 1) issues a request and polls for the response.

The communication is initiated by a xxx.req from the DP-master (Class 1). The MSAC1M passes this request to Layer 2 and checks the status after receipt of the DMPMM1-response. If the response status is NR (no reply data) it will continue with a new SRD with a NULL-PDU to get response data. This procedure will be repeated by the DP-master (Class 1) until

– an erroneous response occurs,

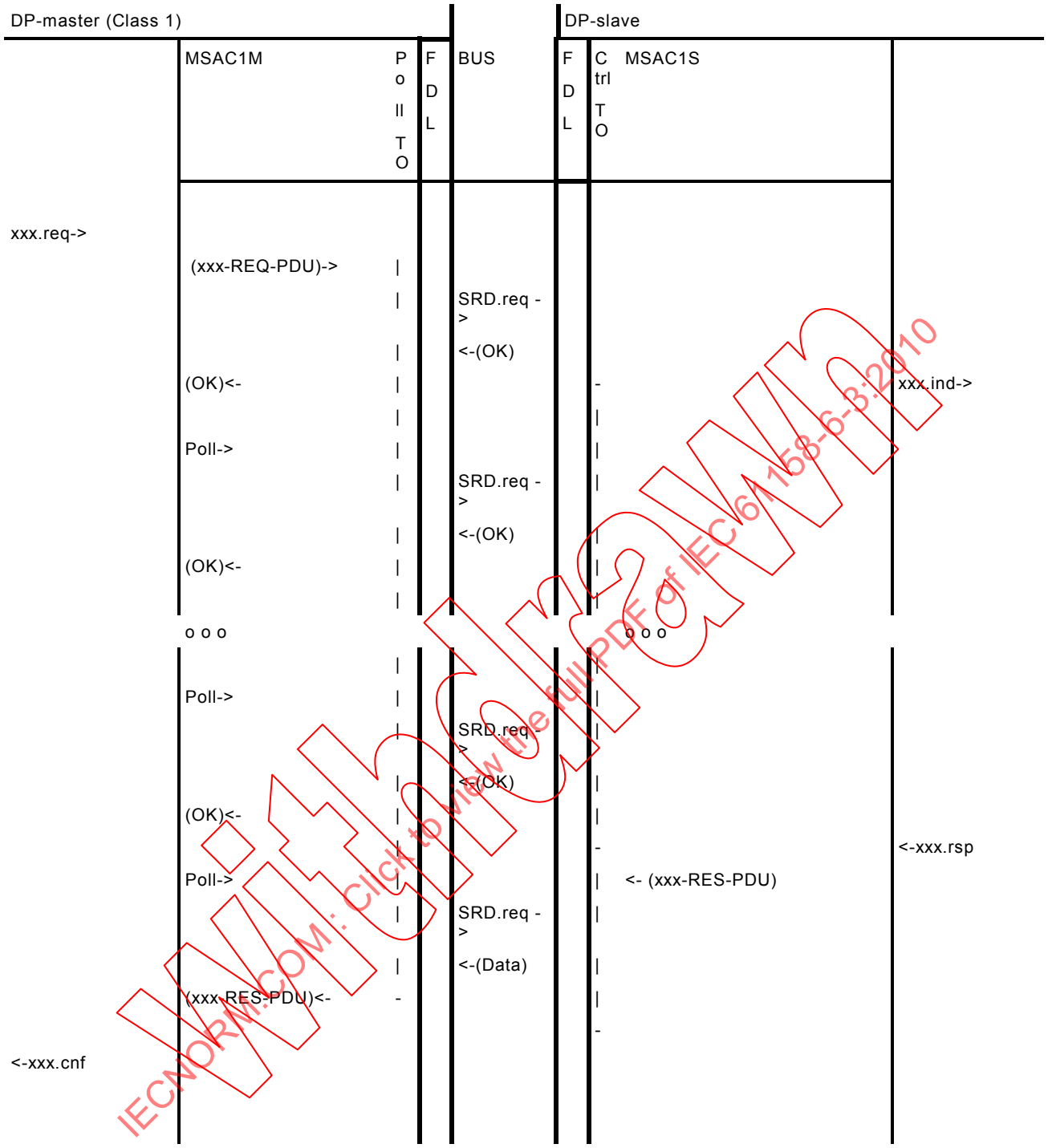
– or a SRD_RES_PDU with response data is received.

The result will then be passed to the User of the DP-master (Class 1) by means of a corresponding xxx.cnf.

The responder (DP-slave) waits for a request. If the MSAC1S of the DP-slave receives a valid request, this request is passed to the User by the corresponding xxx.ind.

After receipt of the response from the User (xxx.rsp) the response data will be passed with a REPLY_UPDATE.req to DMPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn




IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Figure 23 – Sequence of acyclic communication between DP-master (class 1) and DP-slave**

## 6.9 Application relationship monitoring

### 6.9.1 Monitoring of the MS0 – AR

#### 6.9.1.1 General

In industrial control systems there is a need for checking the correct functioning of each individual part. In such a system a medium failure or a defect in a bus master shall not cause errors in the peripheral devices. In the above mentioned cases, a peripheral device shall switch to a safe state. Additionally, the User of a bus master (Class 1) has to be informed about transmission breakdowns after a certain amount of time.

#### 6.9.1.2 Control interval at the DP-slave

**Watchdog control**

This Timer, restarted by received Requests on the bus master side, will set the outputs of a bus slave to the safe state after the expiration of the timer.

**Rules for issuing Check User Prm Result.req and Check Cfg Result.req**

The order of issuing these primitives shall be the same as the order of invocation of Check User Prm.ind and Check Cfg Result.ind primitives by MSCY1S.

#### 6.9.1.3 Control intervals at the DP-master (class 1)

**Min_Slave_Interval**

This monitoring time specifies the smallest allowed period of time between two Slave poll cycles. This ensures that the sequence of function requests from the bus master can be handled by the bus slave. This period of time will be complied by the bus master (Class 1) for every master-slave function with exception of the function Global_Control. For the function Global_Control the User is responsible for compliance with Min_Slave_Interval.

**Data_Control_Time**

The Data_Control_Time defines the time in which a DP-master (Class 1) checks the data transfer to its associated DP-slaves. Furthermore, in half of this time the DP-master (Class 1) indicates his operation mode by means of a Global Control Service to the dedicated DP-slaves .

The Master parameter set contains the Data_Control_Time.

**Rule**

$$T_{WD} > T_{TR} \quad (1)$$

$$T_{WD} > \text{Min_Slave_Interval} \quad (2)$$

$$\text{Data_Control_Time} \geq 6 \cdot T_{WD} \quad (3)$$

### 6.9.2 Monitoring of the MS2 – AR

The objective of the connection monitoring is to detect the failure of the communication partner. Idle PDUs are exchanged to retrigger the monitoring timer of the communication partner.

The Idle service cycle initiated by a bus master is called Master-Idle. The Idle service cycle initiated by the Slave is called Slave-Idle.

All monitoring and idle sequences are controlled with a single timebase “Send Timeout”.

**Send_Timeout**

Each MSAC2 communication relationship is supervised by a Timer (Send_Timeout).

If this Timer expires the bus master (Class 2) aborts the MSAC2-communication relationship. Additionally the User of a bus master (Class 2) has to be informed.

This Timer has to be set up according to

the network load

the number of parallel connections per DP-master

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Rule**

$$T_{Send_TO(TS)} \geq \left(\sum ATCyc\right) \times Num_of_max(TS) \quad \textbf{(4)}$$

where

ATCyc means delay of all activities of a token cycle,

Num_of_max means number of maximal acyclic connections.

NOTE The Send_Timeout parameter in the Initiate service is coded as an unsigned integer value, each unit represents 10ms. Due to Timer errors and the round up of integer, the next possible integer plus one has to be used as Send_Timeout.

It is assumed that one acyclic interaction is executed within a Token cycle at each DP-master. If more than one can be executed, the Send_TO may be divided by the number of acyclic interactions per token cycle.

Figure 24 shows an example for the establishment of an MS2 connection between a Master (Class 2) and a Slave. The connection monitoring starts immediately after this.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

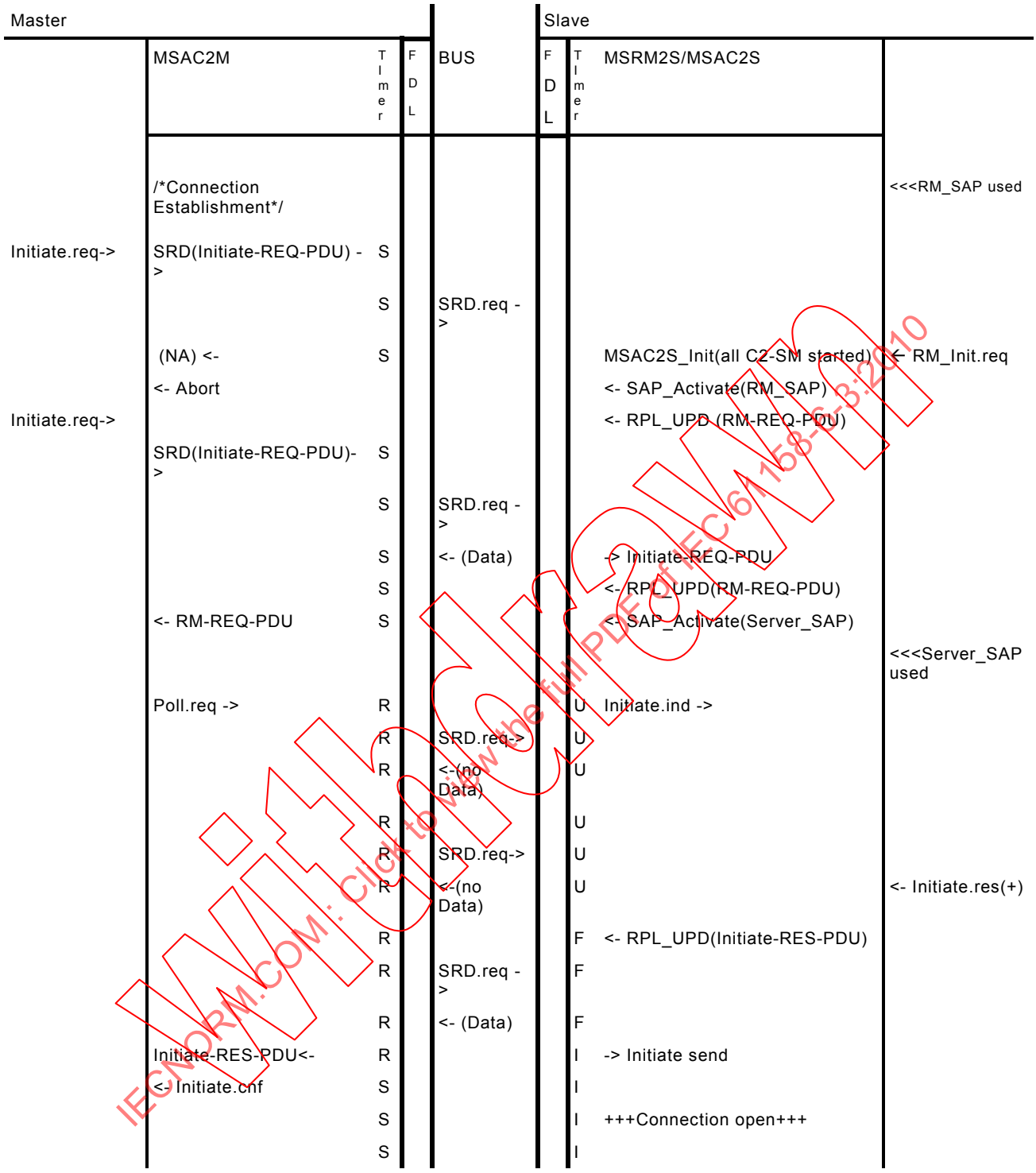



**Figure 24 – Example for connection establishment on MS2**

Phase a) Connection monitoring during pending MS2 service response or running Master-Idle:

The DP-master starts the R-Timer (receive timer) when the MS2 service request is transferred to the local DLL. The R-Timer monitors the remote MSAC2S and the local DLL.

The R-Timer is stopped if a response is received or restarted if a Slave-Idle request is received. The connection monitoring aborts the connection if a negative DL confirmation appears or the R-Timer expires.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

The slave starts the U-Timer (user response timer) when a confirmed service indication is passed to the User. The U-Timer is stopped when the User provides the service response. If the U-Timer expires a Slave-Idle request is sent to retrigger the monitoring R-Timer of the Master.

The DP-slave starts the F-Timer (fetch response timer) when a service response or Slave-Idle request is passed to the local DLL. The F-Timer is stopped when the master has fetched the PDU. If the F-Timer expires the connection monitoring aborts the connection.

The connection monitoring of phase a) is shown in Figure 25.

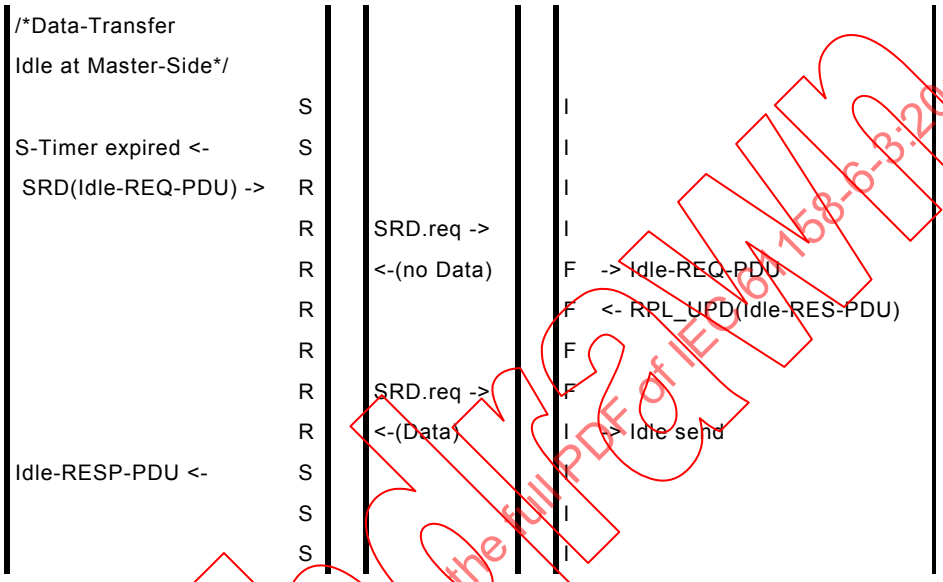



**Figure 25 – Idle at master-side on MS2**

Phase b) Connection monitoring without pending service response:

The bus master starts the S-Timer (send timer) if a response PDU or a Idle-PDU is received. The S-Timer is stopped if a service request is provided by the User. If the S-Timer expires a Master-Idle request is sent to stop the monitoring I-Timer of the Slave.

During pending Master-Idle responses the connection monitoring is handled as described in Phase a).

The Slave starts the I-Timer (indication timer) when the reply update buffer has been fetched by the Master. The I-Timer is stopped if a Master-Idle request or a service indication is received. If the I-Timer expires the connection monitoring aborts the connection.

The Master monitors the transmission of the Abort-REQ-PDU. The Master starts the S-Timer when the Abort.req is transferred to the local DLL. The S-Timer is stopped when the Abort.req is sent. If the S-Timer expires the connection is closed and the User is informed with a MSAC2M_Closed.ind.

The Slave monitors the fetching of the Abort-REQ-PDU by the Master. The Slave starts the F-Timer when the Abort.req is transferred to the local DL. The F-Timer is stopped if the Master has fetched the Abort-REQ-PDU. If the F-Timer expires the Server_SAP is deactivated and the connection is closed.

The connection monitoring of phase b) is shown in Figure 26.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Master | | | | | Slave | | | |
|---|---|---|---|---|---|---|---|---|
| | MSAC2M | Timer | FDL | BUS | FDL | Timer | MSRM2S/MSAC2S | |
| | /*Data-Transfer | | | | | | | |
| | Idle at Slave-Side*/ | | | | | | | |
| Read.req-> | | S | | | | I | | |
| | SRD(Read-REQ-PDU)-> | R | | | | I | | |
| | | R | | SRD.req -> | | I | | |
| | | R | | <-(no Data) | | | -> Read-REQ-PDU | |
| | Poll.req -> | R | | | | U | Read.ind -> | |
| | | R | | SRD.req -> | | U | | |
| | | R | | <-(no Data) | | U | | |
| | | R | | | | U | -> U-Timer expired | |
| | | R | | | | F | <- RPL_UPD(Idle-REQ-PDU) | |
| | | R | | | | F | | |
| | | R | | SRD.req -> | | F | | |
| | | R | | <-(Data) | | I | -> Idle send | |
| | Idle-REQ-PDU<- | R | | | | I | | |
| | SRD(Idle-RES-PDU) -> | R | | | | I | | |
| | | R | | SRD.req -> | | I | | |
| | | R | | <-(no Data) | | U | -> Idle-RES-PDU | |
| | | R | | | | U | | |
| | | R | | SRD.req -> | | U | | <- Read.res |
| | | R | | <-(no Data) | | F | <- RPL_UPD(Read-RES-PDU) | |
| | | R | | | | F | | |
| | | R | | SRD.req -> | | F | | |
| | | R | | <-(Data) | | I | -> Response send | |
| | Read-RES-PDU <- | R | | | | I | | |
| | <- Read.cnf | S | | | | I | | |
| | | S | | | | I | | |

**Figure 26 – Idle at slave-side on MS2**

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# 7 AP-context state machine

There is no AP-Context State Machine defined for this Protocol.

# 8 FAL service protocol machines (FSPMs)

## 8.1 FSPMS

### 8.1.1 Primitive definitions

#### 8.1.1.1 Primitives exchanged between FSPMS and AP-Context

Table 33 shows the service primitives including their associated parameters issued by the AP-Context and received by the FSPMS.

**Table 33 – Primitives issued by AP-Context to FSPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DLL Init DP slave.req | AP-Context | Bus Para | Refer to FAL Service Definition Type 3 Fieldbus, IEC 61158-5-3 |
| SInit MS0.req | AP-Context | AREP | |
| SInit MS1.req | AP-Context | AREP | |
| SInit MS2.req | AP-Context | AREP | |
| Reset DP slave.req | AP-Context | (none) | |
| Abort.req | AP-Context | AREP,<br>Subnet,<br>Instance,<br>Reason Code | |
| DP slave Application Ready.req | AP-Context | AREP | |
| Check User Prm Result.req | AP-Context | AREP,<br>Prm_OK | |
| Check Ext User Prm Result | AP-Context | AREP,<br>Ext_Prm_OK | |
| Check Cfg Result.req | AP-Context | AREP,<br>Cfg_OK,<br>Input Data Len,<br>Output Data Len | |
| Set Cfg.req | AP-Context | AREP,<br>Cfg Data | |
| Set Slave Diag.req | AP-Context | AREP,<br>Ext Diag Overflow,<br>Ext Diag Flag,<br>Ext Diag Data | |
| Set Input.req | AP-Context | AREP,<br>Input Data | |
| Get Output.req | AP-Context | AREP | |
| Alarm Notification.req | AP-Context | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr,<br>Add Ack,<br>Alarm Specifier,<br>Alarm Data | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Initiate.rsp(+) | AP-Context | AREP,<br>Max Len Data Unit,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Initiate.rsp(-) | AP-Context | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Read.rsp(+) | AP-Context | AREP,<br>Length,<br>Data | |
| Read.rsp(-) | AP-Context | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.rsp(+) | AP-Context | AREP,<br>Length | |
| Write.rsp(-) | AP-Context | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Data Transport.rsp(+) | AP-Context | AREP,<br>Length,<br>Data | |
| Data Transport.rsp(-) | AP-Context | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Alarm Ack.rsp(+) | AP-Context | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Alarm Ack.rsp(-) | AP-Context | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Load ARL DP slave.req | AP-Context | Min Send Timeout,<br>List of Resource Manager Entries,<br>List of S_SAP_indices,<br>List of ARL Entries | |
| Get ARL DP slave.req | AP-Context | (none) | |
| Load CRL DP slave.req | AP-Context | List of CRL Entries | |
| Get CRL DP slave.req | AP-Context | (none) | |
| Set ARL Isochronous Mode.req | AP-Context | Isochronous Mode | |
| Get Publisher Data.req | AP-Context | AREP,<br>CREP | |
| Start Subscriber.req | AP-Context | AREP,<br>CREP | |
| Stop Subsciber.req | AP-Context | AREP,<br>CREP | |
| Load CRL DXB-Linktable Entries.req | AP-Context | AREP,<br>DXB-Linktable | |
| Initiate Load.rsp(+) | AP-Context | AREP,<br>Actual LR Size,<br>Max Response Delay,<br>Max Segment Length<br>User Specific | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Initiate Load.rsp(-) | AP-Context | AREP,<br>Error Code | |
| Pull Segment.rsp(+) | AP-Context | AREP,<br>Segment Length,<br>Segment Number,<br>More Follows,<br>Data | |
| Pull Segment.rsp(-) | AP-Context | AREP,<br>Error Code | |
| Push Segment.rsp(+) | AP-Context | AREP | |
| Push Segment.rsp(-) | AP-Context | AREP,<br>Error Code | |
| Terminate Load.rsp(+) | AP-Context | AREP | |
| Terminate Load.rsp(-) | AP-Context | AREP,<br>Error Code | |
| Start.rsp(+) | AP-Context | AREP | |
| Start.rsp(-) | AP-Context | AREP<br>Error Code | |
| Stop.rsp(+) | AP-Context | AREP | |
| Stop.rsp(-) | AP-Context | AREP<br>Error Code | |
| Resume.rsp(+) | AP-Context | AREP | |
| Resume.rsp(-) | AP-Context | AREP<br>Error Code | |
| Reset.rsp(+) | AP-Context | AREP | |
| Reset.rsp(-) | AP-Context | AREP<br>Error Code | |
| Call.rsp(+) | AP-Context | AREP<br>Result Argument | |
| Call.rsp(-) | AP-Context | AREP<br>Error Code | |
| Get FI State.rsp(+) | AP-Context | AREP<br>FI State | |
| Get FI State.rsp(-) | AP-Context | AREP<br>Error Code | |

Table 34 shows the service primitives including their associated parameters issued by the FSPMS and received by the AP-Context.

**Table 34 – Primitives issued by FSPMS to AP-Context**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Sinit DLL.cnf | FSPMS | (none) | Refer to FAL Service Definition Type 3 Fieldbus, IEC 61158-5-3 |
| Sinit MS0.cnf | FSPMS | AREP | |
| Sinit MS1.cnf | FSPMS | AREP | |
| Sinit MS2.cnf | FSPMS | AREP | |
| Reset.cnf | FSPMS | (none) | |
| Check User Prm Result.cnf(+) | FSPMS | AREP | |
| Check User Prm Result.cnf(-) | FSPMS | AREP,<br>Status | |
| Check Ext User Prm Result.cnf(+) | FSPMS | AREP | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Check Ext User Prm Result.cnf(-) | FSPMS | AREP,<br>Status | |
| Check Cfg Result.cnf(+) | FSPMS | AREP | |
| Check Cfg Result.cnf(-) | FSPMS | AREP,<br>Status | |
| Set Cfg.cnf(+) | FSPMS | AREP | |
| Set Cfg.cnf(-) | FSPMS | AREP | |
| Set Slave Diag.cnf | FSPMS | AREP | |
| Set Input.cnf(+) | FSPMS | AREP | |
| Set Input.cnf(-) | FSPMS | AREP | |
| Get Output.cnf(+) | FSPMS | AREP,<br>Output Data,<br>Clear Flag,<br>New Flag | |
| Get Output.cnf(-) | FSPMS | AREP | |
| Started.ind | FSPMS | AREP,<br>Actual Enabled Alarms,<br>Alarm Sequence,<br>Alarm Limit | |
| Stopped.ind | FSPMS | AREP | |
| Abort.ind | FSPMS | AREP,<br>Locally Generated,<br>Subnet,<br>Instance,<br>Reason Code,<br>Additional Detail | |
| Fault.ind | FSPMS | AREP | |
| Alarm Reject.ind | FSPMS | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Set Slave Add.ind | FSPMS | AREP,<br>New Slave Add,<br>Ident Number,<br>No Add Chg,<br>Rem Slave Data | |
| Check User Prm.ind | FSPMS | AREP,<br>User Prm Data | |
| Check Ext User Prm.ind | FSPMS | AREP,<br>Ext User Prm Data | |
| Check Cfg.ind | FSPMS | AREP,<br>Cfg Data | |
| New Output.ind | FSPMS | AREP,<br>Output Data,<br>Clear Flag | |
| Global Control.ind | FSPMS | AREP,<br>Clear Command,<br>Sync Command,<br>Freeze Command,<br>Group Select | |
| Initiate.ind | FSPMS | AREP,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Read.ind | FSPMS | AREP,<br>Slot Number,<br>Index,<br>Length | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Write.ind | FSPMS | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Data Transport.ind | FSPMS | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Alarm Ack.ind | FSPMS | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Load ARL DP slave.cnf(+) | FSPMS | (none) | |
| Load ARL DP slave.cnf(-) | FSPMS | Status | |
| Get ARL DP slave.cnf(+) | FSPMS | Min Send Timeout,<br>List of Resource Manager Entries,<br>List of S_SAP_indices,<br>List of ARL Entries | |
| Get ARL DP slave.cnf(-) | FSPMS | Status | |
| Load CRL DP slave.cnf(+) | FSPMS | (none) | |
| Load CRL DP slave.cnf(-) | FSPMS | Status | |
| Get CRL DP slave.cnf(+) | FSPMS | List of CRL Entries | |
| Get CRL DP slave.cnf(-) | FSPMS | Status | |
| Set ARL Isochronous Mode.cnf(+) | FSPMS | | |
| Set ARL Isochronous Mode.cnf(-) | FSPMS | Status | |
| SYNCH Event.ind | FSPMS | AREP,<br>Status | |
| Publisher Active.ind | FSPMS | AREP,<br>CREP,<br>Status | |
| New Publisher Data.ind | FSPMS | AREP,<br>CREP | |
| Get Publisher Data.cnf(+) | FSPMS | AREP,<br>CREP,<br>Data,<br>New Flag | |
| Get Publisher Data.cnf(-) | FSPMS | AREP,<br>CREP | |
| Start Subsciber.cnf(+) | FSPMS | AREP,<br>CREP | |
| Start Subscriber.cnf(-) | FSPMS | AREP,<br>CREP | |
| Stop Subscriber.cnf(+) | FSPMS | AREP,<br>CREP | |
| Stop Subscriber.cnf(-) | FSPMS | AREP,<br>CREP | |
| Load CRL DXB-Linktable Entries.cnf(+) | FSPMS | AREP | |
| Load CRL DXB-Linktable Entries.cnf(-) | FSPMS | AREP | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Set Time.ind | FSPMS | AREP,<br>Time Value,<br>Local Time Diff,<br>Summertime,<br>Accuracy,<br>Synchronisation Active,<br>Announcement Hour | |
| Sync Interval Violation.ind | FSPMS | AREP | |
| Initiate Load.ind | FSPMS | AREP,<br>Slot Number,<br>LR Index,<br>Load Type,<br>Load Image Size,<br>Intersegment Request Timeout<br>User Specific | |
| Pull Segment.ind | FSPMS | AREP,<br>Slot Number,<br>LR Index,<br>Segment Length | |
| Push Segment.ind | FSPMS | AREP,<br>Slot Number<br>LR Index,<br>Segment Length<br>Segment Number,<br>More Follows,<br>Data | |
| Terminate Load.ind | FSPMS | AREP<br>Slot Number<br>LR Index | |
| Start.ind | FSPMS | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Stop.ind | FSPMS | AREP<br>Slot Number<br>FI Index | |
| Resume.ind | FSPMS | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Reset.ind | FSPMS | AREP<br>Slot Number<br>FI Index | |
| Call.ind | FSPMS | AREP<br>Slot Number<br>Entity Number<br>FI Index<br>Execution Argument | |
| Get FI State | FSPMS | AREP<br>Slot Number<br>FI Index | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 8.1.1.2 Parameters of FSPMS primitives

The parameters used with the primitives exchanged between the FSPMS and the AP-Context are described in FAL Service Definition Type 3 Fieldbus, see IEC 61158-5-3.

### 8.1.2 State machine description

This Machine is used to co-ordinate the Interactions between AP-Context and DMPMS, MSCY1S, MSAC1S, MSRM2S and MSAC2S. As Alarms requires the support of several State Machines the alarms are handled by this machine.

With the SInit_MS1.req, the local variables of the State Machines are set to the start-up values and MSAC1 State Machines is initialised.

When the MSCY1S enters the Data-Exchange-State (This is signalled by a start.ind) alarm messages can be generated by the User.

The parallel processing of alarm messages is limited by:

the number of Alarm_Classes (Alarm_Sequence = False)

the number of Alarms accepted by the Master (Alarm_Sequence = True)

the number of Alarms handled by this Slave (Alarm_Sequence=True).

Alarms may be processed only if the appropriate Alarm_Class is enabled by the DP-master. An Alarm issued by an Alarm_Notification will be transmitted via a diagnosis (MSCY1S) to a DP-master (Class 1). The Acknowledgement will be received by a Alarm Ack service processed by MSAC1S.

The State Machine needs local variables which are described below. Furthermore the machine uses functions and macros which are listed in 8.1.4.

**Local variables**

**Alarm_Sequence**
**(Boolean)**
This variable indicates whether

only one alarm of a specific Alarm_Class can be active at one time (Alarm_Sequence = False), or

several alarms (2 to 32) of any type can be active at one time (Alarm_Sequence = True).

**Initial_Alarm_Sequence**
(Boolean)

This variable stores the basic ability of the Slave.

**Outstanding_Alarm_TABLE**
(Array 0…7, 0…31 of Alarm_Status, Alarm_Type)

Alarm_Status: pending, not_pending

Alarm_Type: 1 to 6, 32 to 126

The Alarm_State_Table is a two dimensional array with 7 * 32 elements. Each element within the array stores information about the actual state of any alarm. The array is addressed with the Alarm_Class calculated from the Alarm_Type and the Seq_Nr of the alarm.

**Alarm_Max**
(Unsigned8)

This variable indicates the maximum number of parallel alarms of this Slave.

Range: 1..32

**Alarm_Limit**
(Unsigned8)

This variable contains the maximum number of Alarms allowed by the actual Master-Slave-connection.

Range: 1..32

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Alarm_Decode**
(Array 0…7 of Unsigned8)

Decoding the number of parallel alarms:

Alarm_Decode[0] = 1

Alarm_Decode[1] = 2

Alarm_Decode[2] = 4

Alarm_Decode[3] = 8

Alarm_Decode[4] = 12

Alarm_Decode[5] = 16

Alarm_Decode[6] = 24

Alarm_Decode[7] = 32

**Alarm_Count**
(Unsigned8)

This counter contains the number of alarms (0..32) which have actually been sent by the DP-slave. The counter is only used in the sequence mode.

Range: 0 to Alarm_Limit

**Req_Alarm_FIFO**
The Req_Alarm_FIFO is used to store the Alarms which are still to be send. Each element consists of an Alarm-PDU. The FIFO shall be able to hold up to 32 respectively Alarm_Max entries.

Actual_Enabled_Alarms

(Array of Bits)

This variable indicates the type of alarms which are actually supported by the Master.

Bit 0 = reserved
Bit 1 = reserved
Bit 2 = Update_Alarm
Bit 3 = Status_Alarm
Bit 4 = Manufacturer_Specific_Alarm
Bit 5 = Diagnostic_Alarm
Bit 6 = Process_Alarm
Bit 7 = Pull_Plug_Alarm

**Alarms_Supported**
(Array of Bits)

This variable indicates the type of alarms which are supported by the Slave.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Bit 0 = reserved

Bit 1 = reserved

Bit 2 = Update_Alarm

Bit 3 = Status_Alarm

Bit 4 = Manufacturer_Specific_Alarm

Bit 5 = Diagnostic_Alarm

Bit 6 = Process_Alarm

Bit 7 = Pull_Plug_Alarm

**Local_Diag_Buffer**
(Octet-String)

This local variable is used to store the Diag-Data except the first 6 Octets and the Alarm- and Status-PDUs.

**L_Ext_Diag_Flag**
(Bit)

This local variable is used to store the Ext_Diag_Flag.

**Stored_Diag**
(Boolean)

Flag which indicates, that new Diag_Data are stored in the Diag_Buffer while waiting for transmission of a previous Diagnosis.

**Diag_Buffer**

(Octet-String)

Buffer for storing a Slave-Diagnostic(Identification-Related-Diagnosis, Channel-Related-Diagnosis, Revision_Number)

**Index**
(Structure of Alarm_Class, Seq_Nr)

This index is used for addressing outstanding Alarm_Table.

**LR_State**
(Unsigned8)

This variable stores the state of the Load Region sequence.

**CurrentBuffer**
(Array of structure Empty, WriteLength and Data)

This buffer stores APDUs for Load Region for each AREP.

**CurrentLoadType**
(Unsigned8)

This variable stores whether a Pull or a Push sequence is active.

**CurrentLRIndex**
(Unsigned16)

This variable stores the LR type.

### 8.1.3 FSPMS state table

Table 35 contains the complete description of the FSPMS state machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Processing constrains for isochronous mode:**
The time from Issuing the SYNCH.ind until the Service Requests from the MSCY1S to the DL shall be small enough to allow the MAC to send out all reguests from MSCY1S within a cycle.

The following definition for a state set is used in Table 35.

t_state = W-START or W-DIA-UPD or W-FETCHED

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 35 – FSPMS state table**

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 1 | POWER-ON | Load ARL DP Slave.req (Min Send Timeout, List of Resource Manager Entries, List of S_SAP_indices, List of ARL Entries)<br>/valid parameters<br>=><br>Load ARL DP Slave.cnf(+) | POWER-ON |
| 2 | POWER-ON | Load ARL DP Slave.req (Min Send Timeout, List of Resource Manager Entries, List of S_SAP_indices, List of ARL Entries)<br>/invalid parameters<br>=><br>Status := NO<br>Load ARL DP Slave.cnf(-) (Status) | POWER-ON |
| 3 | POWER-ON | Get ARL DP Slave.req<br>/ARL loaded<br>=><br>Get ARL DP Slave.cnf(+) ( Min Send Timeout, List of Resource Manager Entries, List of S_SAP_indices, List of ARL Entries ) | POWER-ON |
| 4 | POWER-ON | Get ARL DP Slave.req<br>/ARL not loaded<br>=><br>Status := NO<br>Get ARL DP Slave.cnf(-) (Status) | POWER-ON |
| 5 | POWER-ON | Load CRL DP Slave.req ( List of CRL Entries )<br>/valid parameters<br>=><br>Load CRL DP Slave.cnf(+) | POWER-ON |
| 6 | POWER-ON | Load CRL DP Slave.req ( List of CRL Entries )<br>/invalid parameters<br>=><br>Status := NO<br>Load CRL DP Slave.cnf(-) (Status) | POWER-ON |
| 7 | POWER-ON | Get CRL DP Slave.req<br>/CRL loaded<br>=><br>Get CRL DP Slave.cnf(+) (List of CRL Entries) | POWER-ON |
| 8 | POWER-ON | Get CRL DP Slave.req<br>/CRL not loaded<br>=><br>Status := NO<br>Get CRL DP Slave.cnf(-) (Status) | POWER-ON |
| 9 | POWER-ON | FSPMS_SInit_MS1.req (AREP)<br>/Alarm_Mode_Slave <> 0<br>=><br>Alarm_Sequence := True<br>Initial_Alarm_Sequence := Alarm_Sequence<br>Alarm_Max := Alarm_Decode[Alarm_Mode_Slave]<br>Alarm_Max := max(Alarm_Max, 7)<br>Local_Diag_Buffer := Default_Ext_Diag_Data<br>L_Ext_Diag_Flag := Default_Ext_Diag<br>FILL (Outstanding_Alarm_TABLE.Alarm_Status, not_pending)<br>Stored_Diag:=False<br>Reset_Req_Alarm_FIFO()<br>Act_Ref := 0<br>MSAC1_SInit_MS1.req | W-START-C1 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 10 | POWER-ON | FSPMS_SInit_MS1.req (AREP)<br>/Alarm_Mode_Slave = 0<br>=><br>Alarm_Sequence := False<br>Initial_Alarm_Sequence := Alarm_Sequence<br>Alarm_Max := 7<br>Local_Diag_Buffer := Default_Ext_Diag_Data<br>L_Ext_Diag_Flag := Default_Ext_Diag<br>FILL(Outstanding_Alarm_TABLE.Alarm_Status, not_pending)<br>Stored_Diag:=False<br>Reset_Req_Alarm_FIFO()<br>FSPMS_User_Reset:=False<br>Act_Ref := 0<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSAC1_SInit_MS1.req | W-START-C1 |
| 11 | W-START-C1 | MSAC1S_SInit_MS1.cnf<br>=><br>FSPMS_SInit_MS1.cnf(AREP) | W-START |
| 12 | W-START | FSPMS_Abort.req(AREP)<br>/AREP.AR_Type=MS0<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Abort.req | W-START |
| 13 | W-START | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP, Alarm_Type, Slot_Number, Seq_Nr, Status=No_Start) | W-START |
| 14 | W-START | FSPMS_Set_Slave_Diag.req(AREP,Ext_Diag_Flag, Ext_Diag)<br>=><br>L_Ext_Diag_Flag := Ext_Diag_Flag<br>L_Ext_Diag_Overflow := Ext_Diag_Overflow<br>Local_Diag_Buffer := Ext_Diag<br>Act_Ref := Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference := Act_Ref)<br>FSPMS_Set_Slave_Diag.cnf(AREP) | W-START |
| 15 | W-START | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>=><br>MSCY1S_Abort.req<br>FSPMS_Abort.ind(AREP) | W-START |
| 16 | W-START | MSCY1S_Set_Slave_Diag.cnf(-) (Reference)<br>=><br>ignore | W-START |
| 17 | W-START | MSCY1S_Set_Slave_Diag.cnf(+) (Reference)<br>=><br>ignore | W-START |
| 18 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/(Enabled Alarms & Alarms_Supported) = 0<br>=><br>FSPMS DP Slave Started.ind(AREP,Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | W-START |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 19 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/ ((Enabled Alarms & Alarms_Supported) <> 0) && Alarm_Mode_Master <> 0 && Initial_Alarm_Sequence = TRUE && Diag_Flag=FALSE<br>=><br>Alarm_Sequence := True<br>Actual_Enabled_Alarms := Enabled_Alarms<br>Alarm_Limit := min(Alarm_Max, Alarm_Decode[Alarm_Mode_Master])<br>Alarm_Count := 0<br>FSPMS DP Slave Started.ind(AREP,Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | W-DIA-UPD |
| 20 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/ ((Enabled Alarms & Alarms_Supported) <> 0) && Alarm_Mode_Master <> 0 && Initial_Alarm_Sequence =FALSE<br>=><br>MSCY1S_Abort.req | W-START |
| 21 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/ ((Enabled Alarms & Alarms_Supported) <> 0) && Alarm_Mode_Master = 0 && Diag_Flag=TRUE<br>=><br>Alarm_Sequence := False<br>Actual_Enabled_Alarms := Enabled_Alarms<br>FSPMS DP Slave Started.ind(AREP,Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | W-FETCHED |
| 22 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/ ((Enabled Alarms & Alarms_Supported) <> 0) && Alarm_Mode_Master = 0 && Diag_Flag=FALSE<br>=><br>Alarm_Sequence := False<br>Actual_Enabled_Alarms := Enabled_Alarms<br>FSPMS DP Slave Started.ind(AREP,Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | W-DIA-UPD |
| 23 | W-START | MSCY1S_Stop.ind<br>=><br>FSPMS_Stopped.ind(AREP) | W-START |
| 24 | W-DIA-UPD | FSPMS_Abort.req(AREP)<br>/AREP.AR_Type=MS0<br>=><br>Index := Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Abort.req | RESET-P-LEAVE |
| 25 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=FALSE<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Not_Enabled) | W-DIA-UPD |
| 26 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE && Alarm_Count=Alarm_Limit<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP, Alarm_Type, Slot_Number, Seq_Nr, Status=Limit_Expired) | W-DIA-UPD |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 27 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE &&<br>Alarm_Count<Alarm_Limit<br>&&<br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]=not_pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=pending<br>Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]:=Alarm_Type<br>Alarm_Count:=Alarm_Count+1<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer, Alarm(Slot_Number,Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data))<br>Act_Ref := Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference := Act_Ref)<br>FSPMS_Alarm_Notification.cnf(+)(AREP,Alarm_Type, Slot_Number, Seq_Nr) | W-FETCHED |
| 28 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE &&<br>Alarm_Count<Alarm_Limit<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= pending<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Sequence_Nr_Pending) | W-DIA-UPD |
| 29 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= pending<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Sequence_Nr_Pending) | W-DIA-UPD |
| 30 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= not_pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=pending<br>Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]:=Alarm_Type<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer, Alarm(Slot_Number,Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data))<br>Act_Ref := Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference := Act_Ref)<br>FSPMS_Alarm_Notification.cnf(+)(AREP,Alarm_Type, Slot_Number, Seq_Nr) | W-FETCHED |
| 31 | W-DIA-UPD | FSPMS_Set_Slave_Diag.req(AREP,Ext_Diag_Flag, Ext_Diag)<br>=><br>if ( PrmCmdAck in Ext_Diag or Stored_PrmCmdAck != NULL ) Stored_PrmCmdAck := PrmCmdAck, Ext_Diag.Red_Status := PrmCmdAck endif<br>L_Ext_Diag_Flag := Ext_Diag_Flag<br>L_Ext_Diag_Overflow := Ext_Diag_Overflow<br>Local_Diag_Buffer := Ext_Diag<br>Act_Ref := Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference := Act_Ref)<br>FSPMS_Set_Slave_Diag.cnf(AREP) | W-DIA-UPD |
| 32 | W-DIA-UPD | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Alarm_Sequence=TRUE &&<br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]=pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr] := not_pending<br>Alarm_Count:=Alarm_Count-1<br>FSPMS_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>MSAC1S_Alarm Ack.rsp(+) ( Slot Number, Alarm Type, Seq Nr ) | W-DIA-UPD |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 33 | W-DIA-UPD | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Alarm_Sequence=FALSE &&<br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr] =pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr] :=<br>not_pending<br>FSPMS_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>MSAC1S_Alarm Ack.rsp(+) ( Slot Number, Alarm Type, Seq Nr ) | W-DIA-UPD |
| 34 | W-DIA-UPD | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr] =not_pending<br>=><br>Index := Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>MSCY1S_Abort.req<br>FSPMS_Abort.ind(AREP) | RESET-P-LEAVE |
| 35 | W-DIA-UPD | MSCY1S_Set_Slave_Diag.cnf(-) (Reference)<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 36 | W-DIA-UPD | MSCY1S_Set_Slave_Diag.cnf(+) (Reference)<br>/Reference < Act_Ref<br>=><br>ignore | W-DIA-UPD |
| 37 | — | — | — |
| 38 | W-DIA-UPD | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>=><br>MSCY1S_Abort.req | W-START |
| 39 | W-DIA-UPD | MSCY1S_Stop.ind<br>=><br>Alarm_Count:=0<br>Index := Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending) | RESET-PENDING |
| 40 | RESET-PENDING | /Index.Seq_Nr <> 255<br>=><br>Outstanding Alarm_TABLE.Alarm_Status[Index] := not_pending<br>Alarm_Type := Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]<br>Index := Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number=0,<br>Index.Seq_Nr, Status=MSAL1S_Stopped) | RESET-PENDING |
| 41 | RESET-PENDING | /Index.Seq_Nr = 255<br>=><br>Reset_Req_Alarm_FIFO()<br>FSPMS DP Slave Stopped.ind(AREP) | W-START |
| 42 | RESET-P-LEAVE | /Index.Seq_Nr <> 255<br>=><br>Outstanding Alarm_TABLE.Alarm_Status[Index] := not_pending<br>Alarm_Type:= Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]<br>Index := Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number=0,<br>Index.Seq_Nr, Status=Stopped) | RESET-P-LEAVE |
| 43 | RESET-P-LEAVE | /Index.Seq_Nr = 255<br>=><br>Reset_Req_Alarm_FIFO() | W-START |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 44 | W-FETCHED | FSPMS_Abort.req(AREP)<br>/AREP.AR_Type=MS0 && Alarm_Sequence=TRUE && Alarm_Count=0<br>=><br>Stored_Diag:=False<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer)Act_Ref := Act_Ref + 1<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference := Act_Ref)<br>MSCY1S_Abort.req | W-START |
| 45 | W-FETCHED | FSPMS_Abort.req(AREP)<br>/AREP.AR_Type=MS0 && Alarm_Sequence=FALSE \|\| Alarm_Count<>0<br>=><br>Stored_Diag:=False<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer)<br>Index := Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>Act_Ref := Act_Ref + 1<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference := Act_Ref)<br>MSCY1S_Abort.req | RESET-P-LEAVE |
| 46 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=FALSE<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Not_Enabled) | W-FETCHED |
| 47 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE && Alarm_Count=Alarm_Limit<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Limit_Expired) | W-FETCHED |
| 48 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE && Alarm_Count<Alarm_Limit<br>&&<br>Outstanding_Alarm_TABLE.Alarm_Status[AclsAlarm_Type),Seq_Nr]=not_pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=pending<br>Alarm_Count:=Alarm_Count+1<br>Store_to_Req_Alarm_FIFO(Alarm(Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data))<br>FSPMS_Alarm_Notification.cnf(+)(AREP,Alarm_Type, Slot_Number, Seq_Nr) | W-FETCHED |
| 49 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE && Alarm_Count<Alarm_Limit&&<br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= pending<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Sequence_Nr_Pending) | W-FETCHED |
| 50 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= pending<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP, Alarm_Type, Slot_Number, Seq_Nr, Status=Sequence_Nr_Pending) | W-FETCHED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 51 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[AclsL(Alarm_Type),Seq_Nr]= not_pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[AclsL(Alarm_Type),Seq_Nr] := pending<br>Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr] := Alarm_Type<br>Store_to_Req_Alarm_FIFO(Alarm(Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data))<br>FSPMS_Alarm_Notification.cnf(+)(AREP,Alarm_Type, Slot_Number, Seq_Nr) | W-FETCHED |
| 52 | W-FETCHED | FSPMS_Set_Slave_Diag.req(AREP,Ext_Diag_Flag, Ext_Diag)<br>=><br>if ( PrmCmdAck in Ext_Diag or Stored_PrmCmdAck != NULL ) Stored_PrmCmdAck := PrmCmdAck, Ext_Diag.Red_Status := PrmCmdAck endif<br>Stored_Diag := True<br>L_Ext_Diag_Flag := Ext_Diag_Flag<br>L_Ext_Diag_Overflow := Ext_Diag_Overflow<br>Local_Diag_Buffer := Ext_Diag<br>FSPMS_Set_Slave_Diag.cnf(AREP) | W-FETCHED |
| 53 | W-FETCHED | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Alarm_Sequence=TRUE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls (Alarm_Type),Seq_Nr]=pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=not_pending<br>Alarm_Count:=Alarm_Count-1<br>FSPMS_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>MSAC1S_Alarm Ack.rsp(+) ( Slot Number, Alarm Type, Seq Nr ) | W-FETCHED |
| 54 | W-FETCHED | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]=pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=not_pending<br>FSPMS_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>MSAC1S_Alarm Ack.rsp(+) ( Slot Number, Alarm Type, Seq Nr ) | W-FETCHED |
| 55 | W-FETCHED | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]=not_pending<br>=><br>Index := Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>MSCY1S_Abort.req<br>FSPMS_Abort.ind(AREP) | RESET-P-LEAVE |
| 56 | W-FETCHED | MSCY1S_Set_Slave_Diag.cnf(-) (Reference)<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 57 | W-FETCHED | MSCY1S_Set_Slave_Diag.cnf(+) (Reference)<br>/Reference < Act_Ref<br>=><br>ignore | W-FETCHED |
| 58 | W-FETCHED | MSCY1S_Set_Slave_Diag.cnf(+) (Reference)<br>/Reference >= Act_Ref<br>=><br>Stored_PrmCmdAck := NULL | CHK-TRANSMIT |
| 59 | W-FETCHED | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>=><br>MSCY1S_Abort.req | W-START |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 60 | W-FETCHED | MSCY1S_Stop.ind<br>/Alarm_Sequence=TRUE && Alarm_Count=0<br>=><br>Stored_Diag:=False<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer)<br>Act_Ref := Act_Ref + 1<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag,<br>Reference := Act_Ref)<br>FSPMS_Stopped.ind(AREP) | W-START |
| 61 | W-FETCHED | MSCY1S_Stop.ind<br>/Alarm_Sequence=FALSE \|\| Alarm_Count<>0<br>=><br>Stored_Diag:=False<br>Alarm_Count:=0<br>Index := Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer)<br>Act_Ref := Act_Ref + 1<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag,<br>Reference := Act_Ref) | RESET-PENDING |
| 62 | CHK-TRANSMIT | /Req_Alarm_FIFO_state=empty &&Stored_Diag = FALSE | W-DIA-UPD |
| 63 | CHK-TRANSMIT | /Req_Alarm_FIFO_state=empty && Stored_Diag = TRUE=>Stored_Diag:=FalseExt_Diag_Flag:=L_Ext_Diag_FlagExt_Diag_Overflow:= L_Ext_Diag_OverflowExt_Diag:=(Local_Diag_Buffer)Act_Ref := Act_Ref + 1MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference := Act_Ref) | W-FETCHED |
| 64 | CHK-TRANSMIT | /Req_Alarm_FIFO_state<>empty<br>=><br>Alarm:=Load_from_Req_Alarm_FIFO()<br>Stored_Diag:=False<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer, Alarm)<br>Act_Ref := Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag,<br>Reference := Act_Ref) | W-FETCHED |
| 65 | W-RESET | DMPMS_Reset.cnf<br>/R_Cnt<3<br>=><br>R_Cnt:=R_Cnt+1 | W-RESET |
| 66 | W-RESET | MSCY1S_Reset.cnf<br>/R_Cnt<3<br>=><br>R_Cnt:=R_Cnt+1 | W-RESET |
| 67 | W-RESET | MSAC1S_Reset.cnf<br>/R_Cnt<3<br>=><br>R_Cnt:=R_Cnt+1 | W-RESET |
| 68 | W-RESET | MSRM2S_Reset.cnf<br>/R_Cnt<3<br>=><br>R_Cnt:=R_Cnt+1 | W-RESET |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 69 | W-RESET | DMPMS_Reset.cnf<br>/R_Cnt>=3<br>=><br>if (FSPMS_User_Reset=True)<br>FSPMS_Reset DP Slave .cnf<br>endif | POWER-ON |
| 70 | W-RESET | MSCY1S_Reset.cnf<br>/R_Cnt>=3<br>=><br>if (FSPMS_User_Reset=True)<br>FSPMS_Reset DP Slave .cnf<br>endif | POWER-ON |
| 71 | W-RESET | MSAC1S_Reset.cnf<br>/R_Cnt>=3<br>=><br>if (FSPMS_User_Reset=True)<br>FSPMS_Reset DP Slave .cnf<br>endif | POWER-ON |
| 72 | W-RESET | MSRM2S_Reset.cnf<br>/R_Cnt>=3<br>=><br>if (FSPMS_User_Reset=True) FSPMS_Reset DP Slave .cnfendif | POWER-ON |
| 73 | any state | DMPMS_SYCL_Clock_Value.ind (Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg)<br>=><br>FSPMS_SYCL_Clock_Value.ind (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg) | SAME |
| 74 | any state | DMPMS_Fault.ind<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 75 | any state | MSAC1S_Fault.ind<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 76 | any state | MSCY1S_Fault.ind<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 77 | any state | MSRM2S_Fault.ind<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 78 | any state | FSPMS_Reset DP Slave .req<br>=><br>R_Cnt:=0<br>FSPMS_User_Reset:=True<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 79 | W-FETCHED | Set_ARL_Isochron_Mode.req(Isochron_Mode)<br>=><br>Set_ARL_Isochron_Mode.cnf(+) | W-FETCHED |
| 80 | W-DIA-UPD | Set_ARL_Isochron_Mode.req(Isochron_Mode)<br>=><br>Set_ARL_Isochron_Mode.cnf(+) | W-DIA-UPD |
| 81 | W-START | Set_ARL_Isochron_Mode.req(Isochron_Mode)<br>=><br>Status:= NO<br>Set_ARL_Isochron_Mode.cnf(-)(Status) | W-START |
| 82 | t-state | Load CRL DXB Linktable Entries.req(AREP, DXB-Linktable)<br>/valid Parameters<br>=><br>Load CRL DXB Linktable Entries.cnf(+) | |
| 83 | t-state | Load CRL DXB Linktable Entries.req(AREP, DXB-Linktable)<br>/invalid Parameters<br>=><br>Status:= NO<br>Load CRL DXB Linktable Entries.cnf(-) (Status) | |
| 84 | t-state | FSPMS DLL Init DP Slave.req ( Bus Para )<br>=><br>DMPMS_Sinit_DLL.req ( Bus_Para ) | SAME |
| 85 | t-state | FSPMS_SInit MS0.req ( AREP)<br>=><br>MSCY1S_SInit MS0.req ( ) | SAME |
| 86 | t-state | FSPMS_SInit MS2.req ( )<br>=><br>MSRM2S_SInit MS2.req ( ) | SAME |
| 87 | t-state | FSPMS_Abort.req ( AREP, Subnet, Instance, Reason Code )<br>/AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSAC2S_Abort.req ( Subnet, Instance, Reason Code ) | SAME |
| 88 | t-state | FSPMS_DP Slave Application Ready.req ( AREP )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Application Ready.req ( ) | SAME |
| 89 | t-state | FSPMS_Check User Prm Result.req ( AREP, Prm_OK )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Check User Prm Result.req ( Prm_OK ) | SAME |
| 90 | t-state | FSPMS_Check Cfg Result.req ( AREP, Cfg_OK, Input Data Len, Output Data Len )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len ) | SAME |
| 91 | t-state | FSPMS_Set Cfg.req ( AREP, Cfg Data )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Set Cfg.req ( Cfg Data ) | SAME |
| 92 | t-state | FSPMS_Set Input.req ( AREP, Input Data )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Set Input.req ( Input Data ) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 93 | t-state | FSPMS_Get Output.req ( AREP )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Get Output.req ( ) | SAME |
| 94 | t-state | FSPMS_Initiate.rsp(+) ( AREP, Max Len Data Unit, Features Supported, Profile Features Supported, Profile Ident Number, Add Addr Param )<br>/AREP.AR_Type=MS1<br>=><br>MSAC2S_Initiate.rsp(+) ( Res SAP, Max Len Data Unit, Features Supported, Profile Features Supported, Profile Ident Number, Add Addr Param ) | SAME |
| 95 | t-state | FSPMS_Initiate.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Initiate.rsp(-) ( Res SAP, Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 96 | t-state | FSPMS_Read.rsp(+) ( AREP, Length, Data )<br>/AREP.AR_Type=MS1<br>=><br>MSAC1S_Read.rsp(+) ( Length, Data ) | SAME |
| 97 | t-state | FSPMS_Read.rsp(+) ( AREP, Length, Data )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Read.rsp(+) ( Length, Data ) | SAME |
| 98 | t-state | FSPMS_Read.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS1<br>=><br>MSAC1S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 99 | t-state | FSPMS_Read.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 100 | t-state | FSPMS_Write.rsp(+) ( AREP, Length )<br>/AREP.AR_Type=MS1<br>=><br>MSAC1S_Write.rsp(+) ( Length ) | SAME |
| 101 | t-state | FSPMS_Write.rsp(+) ( AREP, Length )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Write.rsp(+) ( Length ) | SAME |
| 102 | t-state | FSPMS_Write.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS1<br>=><br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 103 | t-state | FSPMS_Write.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 104 | t-state | FSPMS_Data Transport.rsp(+) ( AREP, Length, Data )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Data Transport.rsp(+) ( Res SAP, Length, Data ) | SAME |
| 105 | t-state | FSPMS_Data Transport.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Data Transport.rsp(-) ( Res SAP, Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 106 | t-state | FSPMS_Alarm Ack.rsp(+) ( AREP, Slot Number, Alarm Type, Seq Nr )<br>/AREP.AR_Type=MS1<br>=> | SAME |
| 107 | t-state | FSPMS_Alarm Ack.rsp(-) ( AREP, Slot Number, Alarm Type, Seq Nr )<br>/AREP.AR_Type=MS1<br>=> | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 108 | t-state | DMPMS_SInit DLL.cnf ( )<br>=><br>FSPMS DLL Init DP Slave.cnf ( ) | SAME |
| 109 | t-state | MSCY1S_SInit MS0.cnf ( )<br>=><br>AREP=MS0-AR<br>FSPMS_SInit MS0.cnf ( AREP ) | SAME |
| 110 | t-state | MSRM2S_SInit MS2.cnf ( )<br>=><br>FSPMS_SInit MS2.cnf ( ) | SAME |
| 111 | t-state | MSCY1S_Check User Prm Result.cnf(+) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Check User Prm Result.cnf(+) ( AREP ) | SAME |
| 112 | t-state | MSCY1S_Check User Prm Result.cnf(-) ( Status )<br>=><br>AREP=MS0-AR<br>FSPMS_Check User Prm Result.cnf(-) ( AREP, Status ) | SAME |
| 113 | t-state | MSCY1S_Check Cfg Result.cnf(+) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Check Cfg Result.cnf(+) ( AREP ) | SAME |
| 114 | t-state | MSCY1S_Check Cfg Result.cnf(-) ( Status )<br>=><br>AREP=MS0-AR<br>FSPMS_Check Cfg Result.cnf(-) ( AREP, Status ) | SAME |
| 115 | t-state | MSCY1S_Set Cfg.cnf(+) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Cfg.cnf(+) ( AREP ) | SAME |
| 116 | t-state | MSCY1S_Set Cfg.cnf(-) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Cfg.cnf(-) ( AREP ) | SAME |
| 117 | t-state | MSCY1S_Set Input.cnf(+) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Input.cnf(+) ( AREP ) | SAME |
| 118 | t-state | MSCY1S_Set Input.cnf(-) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Input.cnf(-) ( AREP ) | SAME |
| 119 | t-state | MSCY1S_Get Output.cnf(+) ( Output Data, Clear Flag, New Flag )<br>=><br>AREP=MS0-AR<br>FSPMS_Get Output.cnf(+) ( AREP, Output Data, Clear Flag, New Flag ) | SAME |
| 120 | t-state | MSCY1S_Get Output.cnf(-) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Get Output.cnf(-) ( AREP ) | SAME |
| 121 | t-state | MSCY1S_Set Slave Add.ind ( New Slave Add, Ident Number, No Add Chg, Rem Slave Data )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Slave Add.ind ( AREP, New Slave Add, Ident Number, No Add Chg, Rem Slave Data ) | SAME |
| 122 | t-state | MSCY1S_Check User Prm.ind ( User Prm Data )<br>=><br>AREP=MS0-AR<br>FSPMS_Check User Prm.ind ( AREP, User Prm Data ) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 123 | t-state | MSCY1S_Check Cfg.ind ( Check Cfg Mode, Cfg Data )<br>=><br>AREP=MS0-AR<br>FSPMS_Check Cfg.ind ( AREP, Check Cfg Mode, Cfg Data ) | SAME |
| 124 | t-state | MSCY1S_New Output.ind ( Clear Flag )<br>=><br>AREP=MS0-AR<br>FSPMS_New Output.ind ( AREP, Clear Flag ) | SAME |
| 125 | t-state | MSCY1S_Global Control.ind ( Clear Command, Sync Command, Freeze Command, Group Select )<br>=><br>AREP=MS0-AR<br>FSPMS_Global Control.ind ( AREP, Clear Command, Sync Command, Freeze Command, Group Select ) | SAME |
| 126 | t-state | MSAC2S_Initiate.ind ( Res SAP, Features Supported, Profile Features Supported, Profile Ident Number, Add Addr Param )<br>/SetContext(Res_SAP, Service)=TRUE<br>=><br>FSPMS_Initiate.ind ( AREP, Features Supported, Profile Features Supported, Profile Ident Number, Add Addr Param ) | SAME |
| 127 | t-state | MSAC2S_Abort.ind ( Res SAP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)<br>/SetContext(Res_SAP, Service)=TRUE<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>FSPMS_Abort.ind ( AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) | SAME |
| 128 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/SetContext(Res_SAP, Service)=TRUE &&<br>Index<>255<br>=><br>FSPMS_Read.ind ( AREP, Slot Number, Index, Length ) | SAME |
| 129 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/SetContext(Res_SAP, Service)=TRUE &&<br>Index<>255<br>=><br>FSPMS_Read.ind ( AREP, Slot Number, Index, Length ) | SAME |
| 130 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/SetContext(Res_SAP, Service)=TRUE &&<br>Index<>255<br>=><br>FSPMS_Write.ind ( AREP, Slot Number, Index, Length, Data ) | SAME |
| 131 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/SetContext(Res_SAP, Service)=TRUE &&Index<>255<br>=><br>FSPMS_Write.ind ( AREP, Slot Number, Index, Length, Data ) | SAME |
| 132 | t-state | MSAC2S_Data Transport.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/SetContext(Res_SAP, Service)=TRUE<br>=><br>FSPMS_Data Transport.ind ( AREP, Slot Number, Index, Length, Data ) | SAME |
| 133 | t-state | FSPMS_Get_Publisher_Data.req ( AREP, CREP )<br>/AREP.AR_Type=MS0<br>=><br>Rem_Add=CREP.Rem_Add<br>SSCY1S_Get_Publisher_Data.req (Rem_Add) | SAME |
| 134 | t-state | FSPMS_Start_Subscriber.req ( AREP, CREP )<br>/AREP.AR_Type=MS0<br>=><br>Rem_Add=CREP.Rem_Add<br>SSCY1S_Start_Subscriber.req (Rem_Add) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 135 | t-state | FSPMS_Stop_Subscriber.req ( AREP, CREP )<br>/AREP.AR_Type=MS0<br>=><br>Rem_Add=CREP.Rem_Add<br>SSCY1S_Stop_Subscriber.req (Rem_Add) | SAME |
| 136 | t-state | FSPMS_Check_Ext_User_Prm_Result.req ( AREP, Ext_Prm_OK )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Check_Ext_User_Prm_Result.req (Ext_Prm_OK) | SAME |
| 137 | t-state | SSCY1S_Get_Publisher_Data.cnf(+) (Rem_Add, Data, New_Flag)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Get_Publisher_Data.cnf(+) (AREP, CREP, Data, New_Flag) | SAME |
| 138 | t-state | SSCY1S_Get_Publisher_Data.cnf(-) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Get_Publisher_Data.cnf(-) (AREP, CREP) | SAME |
| 139 | t-state | SSCY1S_Start_Subscriber.cnf(+) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Start_Subscriber.cnf(+) (AREP, CREP) | SAME |
| 140 | t-state | SSCY1S_Start_Subscriber.cnf(-) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Start_Subscriber.cnf(-) (AREP, CREP) | SAME |
| 141 | t-state | SSCY1S_Stop_Subscriber.cnf(+) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Stop_Subscriber.cnf(+) (AREP, CREP) | SAME |
| 142 | t-state | SSCY1S_Stop_Subscriber.cnf(-) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Stop_Subscriber.cnf(-) (AREP, CREP) | SAME |
| 143 | t-state | SSCY1S_New_Publisher_Data.ind (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_New_Publisher_Data.ind(AREP, CREP) | SAME |
| 144 | t-state | SSCY1S_Publisher_Active.ind (Rem_Add, Status)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Publisher_Active.ind(AREP, CREP, Status) | SAME |
| 145 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.IL.Extended_Function_Num=Initiate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>Current Load Type[AREP]:=Data.w.IL.Load_Type<br>LR Index:=Data.w.IL.LR_Index<br>Load Type:=Data.w.IL.Load_Type<br>Load Image Size:=Data.w.IL.Load_Image_Size<br>User Specific:=Data.w.IL.User_Specific<br>Intersegment Request Timeout:=Data.w.IL.Intersegment_Request_Timeout<br>FSPMS_Initiate_Load.ind ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>LR_State[AREP]:=W-LR-RES | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 146 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.IL.Extended_Function_Num=Initiate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>Current Load Type[AREP]:=Data.w.IL.Load_Type<br>LR Index:=Data.w.IL.LR_Index<br>Load Type:=Data.w.IL.Load_Type<br>Load Image Size:=Data.w.IL.Load_Image_Size<br>User Specific:=Data.w.IL.User_Specific<br>Intersegment Request Timeout:=Data.w.IL.Intersegment_Request_Timeout<br>FSPMS_Initiate_Load.ind ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 147 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.IL.Extended_Function_Num=Initiate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>CurrentLoadType[AREP]:=Data.w.IL.Load_Type<br>CurrentLRIndex[AREP]:=Data.w.IL.LR_Index<br>LR Index:=Data.w.IL.LR_Index<br>Load Type:=Data.w.IL.Load_Type<br>Load Image Size:=Data.w.IL.Load_Image_Size<br>User Specific:=Data.w.IL.User_Specific<br>Intersegment Request Timeout:=Data.w.IL.Intersegment_Request_Timeout<br>FSPMS_Initiate_Load.ind ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 148 | t-state | FSPMS_Initiate_Load.rsp(+) ( AREP, Actual LR Size, Max Response Delay, Max Segment Length, User Specific )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].LR:=TRUE<br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.IL.Actual_LR_Size:=Actual LR Size<br>CurrentBuffer[AREP].data.r.IL.Max_Response_Delay:=Max Response Delay<br>CurrentBuffer[AREP].data.r.IL.Max_Segment_Length:=Max Segment Length<br>CurrentBuffer[AREP].data.r.IL.User_Specific:=User Specific<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 149 | t-state | FSPMS_Initiate_Load.rsp(+) ( AREP, Actual LR Size, Max Response Delay, Max Segment Length, User Specific )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].LR:=TRUE<br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.IL.Actual_LR_Size:=Actual LR Size<br>CurrentBuffer[AREP].data.r.IL.Max_Response_Delay:=Max Response Delay<br>CurrentBuffer[AREP].data.r.IL.Max_Segment_Length:=Max Segment Length<br>CurrentBuffer[AREP].data.r.IL.User_Specific:=User Specific<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 150 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )/LR_State[AREP]==W-LR-IND&SetContext(Res_SAP, Service)=TRUE &&(Index=255 && CurrentBuffer[AREP].Empty=FALSE && CurrentBuffer[AREP].LR=FALSE )=>Data:=CurrentBuffer[AREP].dataLength:=sizeof(CurrentBuffer[AREP].data)MSAC1S_Read.rsp(+) ( Length, Data )LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 151 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=FALSE &&<br>CurrentBuffer[AREP].LR=TRUE )<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Data:=CurrentBuffer[AREP].data<br>Length:=sizeof(CurrentBuffer[AREP].data)<br>MSAC1S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 152 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=FALSE &&<br>CurrentBuffer[AREP].LR=FALSE )<br>=><br>Data:=CurrentBuffer[AREP].data<br>Length:=sizeof(CurrentBuffer[AREP].data)<br>MSAC2S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 153 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=FALSE &&<br>CurrentBuffer[AREP].LR=TRUE )<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Data:=CurrentBuffer[AREP].data<br>Length:=sizeof(CurrentBuffer[AREP].data)<br>MSAC2S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 154 | t-state | FSPMS_Initiate_Load.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 155 | t-state | FSPMS_Initiate_Load.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 156 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE<br>&&(Index=255 && CurrentBuffer[AREP].Empty=TRUE &&<br>CurrentBuffer[AREP].LR=TRUE )<br>=><br>LR Index:=CurrentLRIndex[AREP]<br>Segment Length:=Length<br>FSPMS_Pull_Segment.ind ( AREP, Slot Number, LR Index, Segment Length )<br>LR_State[AREP]:=W-LR-RES | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 157 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=TRUE &&<br>CurrentBuffer[AREP].LR=FALSE )<br>=><br>Error Decode:=DPV1<br>Error Code 1 :=state conflict<br>Error Code 2:=0<br>MSAC1S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 158 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=TRUE &&<br>CurrentBuffer[AREP].LR=TRUE )<br>=><br>LR Index:=CurrentLRIndex[AREP]<br>Segment Length:=Length<br>FSPMS_Pull_Segment.ind ( AREP, Slot Number, LR Index, Segment Length )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 159 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=TRUE &&<br>CurrentBuffer[AREP].LR=FALSE )<br>=><br>Error Decode:=DPV1<br>Error Code 1 :=state conflict<br>Error Code 2:=0<br>MSAC2S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 160 | t-state | FSPMS_Pull_Segment.rsp(+) ( AREP, Segment Length, Segment Number, More Follows, Data )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Data.PULL.Extended_Function_Num=Pull<br>Data.PULL.Sequence Number:=Segment Number<br>Data.PULL.Options.More Follows:=More Follows<br>Data.PULL.Region Data := Data<br>Length:=6+Segment Length<br>MSAC1S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 161 | t-state | FSPMS_Pull_Segment.rsp(+) ( AREP, Segment Length, Segment Number, More Follows, Data )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Data.PULL.Extended_Function_Num=Pull<br>Data.PULL.Sequence Number:=Segment Number<br>Data.PULL.Options.More Follows:=More Follows<br>Data.PULL.Region Data := Data<br>Length:=6+Segment Length<br>MSAC2S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 162 | t-state | FSPMS_Pull Segment.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 163 | t-state | FSPMS_Pull Segment.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC2S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 164 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.PUSH.Extended_Function_Num=Push)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>LR Index:=Data.PUSH.LR_Index<br>Segment Length:=Length<br>Segment Number:=Data.PUSH.Segment_Number<br>More Follows:=Data.PUSH.More_Follows<br>Data:=Data.PUSH.data<br>FSPMS_Push_Segment.ind ( AREP, Slot Number, LR Index, Segment Length,<br>Segment Number, More Follows, Data )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 165 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.PUSH.Extended_Function_Num=Push)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>LR Index:=Data.PUSH.LR_Index<br>Segment Length:=Length<br>Segment Number:=Data.PUSH.Segment_Number<br>More Follows:=Data.PUSH.More_Follows<br>Data:=Data.PUSH.data<br>FSPMS_Push_Segment.ind ( AREP, Slot Number, LR Index, Segment Length,<br>Segment Number, More Follows, Data )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 166 | t-state | FSPMS_Push_Segment.rsp(+) ( AREP)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 167 | t-state | FSPMS_Push_Segment.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 168 | t-state | FSPMS_Push_Segment.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 169 | t-state | FSPMS_Push_Segment.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 170 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.TL.Extended_Function_Num=Terminate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>LR Index:=Data.w.TL.LR_Index<br>FSPMS_Terminate_Load.ind ( AREP, Slot Number, LR Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 171 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.TL.Extended_Function_Num=Terminate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>LR Index:=Data.w.TL.LR_Index<br>FSPMS_Terminate_Load.ind ( AREP, Slot Number, LR Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 172 | t-state | FSPMS_Terminate_Load.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].LR:=FALSE<br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r:=NIL<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 173 | t-state | FSPMS_Terminate_Load.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].LR:=FALSE<br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r:=NIL<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 174 | t-state | FSPMS_Terminate_Load.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 175 | t-state | FSPMS_Terminate_Load.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 176 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&CurrentBuffer[AREP].LR:=FALSE &&SetContext(Res_SAP, Service)=TRUE<br>&&(Index=255 && Length >= 4) &&(Data.w.CALL.Extended_Function_Num=CALL)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br><br>FI Index:=Data.w.CALL.FI_Index<br>Entity_Number :=Data.w.CALL. Entity_Number<br>Execution Argument:=Data.w.CALL.Execution_Argument<br>FSPMS_Call.ind ( AREP, Slot Number, Entity_Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 177 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&CurrentBuffer[AREP].LR:=True &&<br>SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.CALL.Extended_Function_Num=CALL)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=state conflict<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 178 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&CurrentBuffer[AREP].LR:=FALSE &&<br>SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.CALL.Extended_Function_Num=CALL)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br><br>Entity_Number :=Data.w.CALL. Entity_Number<br>FI Index:=Data.w.CALL.FI_Index<br>Execution Argument:=Data.w.CALL.Execution_Argument<br>FSPMS_Call.ind ( AREP, Slot Number, Entity_Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 179 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&CurrentBuffer[AREP].LR:=True &&<br>SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.CALL.Extended_Function_Num=CALL)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=state conflict<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 180 | t-state | FSPMS_Call.rsp(+) ( AREP, Result Argument )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.CALL.ResultArgument:=Result Argument<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 181 | t-state | FSPMS_Call.rsp(+) ( AREP, Result Argument )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.CALL.ResultArgument:=Result Argument<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 182 | t-state | FSPMS_Call.rsp(-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 183 | t-state | FSPMS_Call.rsp(-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 184 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Start)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>Execution Argument:=Data.FIS.Execution_Argument<br>FSPMS_Start.ind ( AREP, Slot Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 185 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Start)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>Execution Argument:=Data.FIS.Execution_Argument<br>FSPMS_Start.ind ( AREP, Slot Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 186 | t-state | FSPMS_Start.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 187 | t-state | FSPMS_Start.rsp(+) ( AREP )/LR_State[AREP]==W-LR-RES&AREP.AR_Type=MS2=>Length:=CurrentBuffer[AREP].WriteLengthMSAC2S_Write.rsp(+) ( Length )LR_State[AREP]:=W-LR-IND | SAME |
| 188 | t-state | FSPMS_Start.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 189 | t-state | FSPMS_Start.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 190 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Stop)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>FSPMS_Stop.ind ( AREP, Slot Number, FI Index)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 191 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Stop)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>FSPMS_Stop.ind ( AREP, Slot Number, FI Index)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 192 | t-state | FSPMS_Stop.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 193 | t-state | FSPMS_Stop.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 194 | t-state | FSPMS_Stop.rsp(-) ( AREP, Error Code )/LR_State[AREP]==W-LR-RES&AREP.AR_Type=MS1=>CurrentBuffer[AREP].Empty:=TRUEError Decode:=DPV1Error Code 1 :=Error CodeError Code 2:=0MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 195 | t-state | FSPMS_Stop.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 196 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Resume)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>Execution Argument:=Data.FIS.Execution_Argument<br>FSPMS_Resume.ind ( AREP, Slot Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 197 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Resume)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>Execution Argument:=Data.FIS.Execution_Argument<br>FSPMS_Resume.ind ( AREP, Slot Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 198 | t-state | FSPMS_Resume.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 199 | t-state | FSPMS_Resume.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 200 | t-state | FSPMS_Resume.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 201 | t-state | FSPMS_Resume.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 202 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Reset)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>FSPMS_Reset.ind ( AREP, Slot Number, FI Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 203 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Reset)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>FSPMS_Reset.ind ( AREP, Slot Number, FI Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 204 | t-state | FSPMS_Reset.rsp (+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 205 | t-state | FSPMS_Reset.rsp (+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 206 | t-state | FSPMS_Reset.rsp (-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 207 | t-state | FSPMS_Reset.rsp (-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 208 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.STATE.Extended_Function_Num=Get_State)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.w.STATE.FI_Index<br>FSPMS_Get_FI_State.ind ( AREP, Slot Number, FI Index )<br>LR_State[AREP]:=W-LR-RES | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 209 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.STATE.Extended_Function_Num=Get_State)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.w.STATE.FI_Index<br>FSPMS_Get_FI_State.ind ( AREP, Slot Number, FI Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 210 | t-state | FSPMS_Get_FI_State.rsp (+) ( AREP, FI State )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.STATE.FI_State:=FI State<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 211 | t-state | FSPMS_Get_FI_State.rsp (+) ( AREP, FI State )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.STATE.FI_State:=FI State<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 212 | t-state | FSPMS_Get_FI_State.rsp(-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 213 | t-state | FSPMS_Get_FI_State.rsp(-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1 :=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 8.1.4 Functions

Table 36 contains the functions used by the FSPMS, their arguments and their descriptions.

**Table 36 – Functions used by the FSPMS**

| Function name | Description |
|---|---|
| SetContext(Rem_Add, Service) | This function checks if there exist an entry for the inputs Rem_Add resp. Res_SAP and invoked Service in the ARL and related CRL of the DP-slave.<br>A) If there exist an entry<br>then it returns TRUE and sets the local context according to the current<br>CREP and AREP.<br>B) Otherwise it returns FALSE. |
| FILL(Outstanding_Alarm_TABLE.Alarm_Status, State) | This function sets each entry of the two dimensional Outstanding_Alarm_TABLE to the given initial value of State. Valid parameters for State are pending/not_pending. The function has no return value. |
| Search_In_Outstanding_Alarm_TABLE(Alarm_Status) | This function scans each entry of the two dimensional Outstanding_Alarm_TABLE for a specific Alarm_Status. The return value of this function is an index that refers to an Alarm_Table entry that fulfills the Alarm_Status criteria pending/not_pending. |
| SReset_Req_Alarm_FIFO() | This function resets the FIFO for alarms not yet sent. During this function all entries are cleared. The function has no return value. |
| Store_To_Req_Alarm_FIFO(Alarm) | By means of this function the Alarm is stored into the Req_Alarm_FIFO. The function has no return value. |
| Load_From_Req_Alarm_FIFO () | By means of this function one Alarm is read from the Req_Alarm_FIFO. The return value is an Alarm. |
| Req_Alarm_FIFO_state () | This function checks the FIFO for alarms not yet sent. The function has the return values empty/not_empty. |
| AcIs(Alarm_Type) | This function calculates the Alarm_Type related index as return value as follows:<br>if (Alarm_Type = 1) return 0<br>if (Alarm_Type = 2) return 1<br>if (Alarm_Type = 3) return 2<br>if (Alarm_Type = 4) return 3<br>if (Alarm_Type = 5) return 4<br>if (Alarm_Type = 6) return 5<br>if (Alarm_Type >= 32) && (Alarm_Type <= 126) return 6 |
| IsDynamicAttributeLoadable(Attribute) | This function checks the ARL and CRL if the given attribute is loadable in the current state.<br>For the attribute DXB-Linktable the function returns TRUE: if all elements in the DXB-Linktable have a corresponding attribute in the CRL and non of the corresponding SSCY1S statemachines is in the state RUN. |
| SetDynamicAtrribute(Attribute) | This function stores the given attribute in the ARL or CRL. For the attribute DXB-Linktable each entry is stored at the corrosponding CRL attribute. |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Macros**

**ALARM_ENABLED**

(
((Alarm_Type=3 OR Alarm_Type=4) &&
Actual_Enabled_Alarms[7]=TRUE)
OR ((Alarm_Type=2) && Actual_Enabled_Alarms[6]=TRUE)
OR ((Alarm_Type=1) && Actual_Enabled_Alarms[5]=TRUE)
OR ((Alarm_Type>=32 && Alarm_Type<=126)
&& Actual_Enabled_Alarms[4]=TRUE)
OR ((Alarm_Type=5) && Actual_Enabled_Alarms[3]=TRUE)

OR ((Alarm_Type=6) && Actual_Enabled_Alarms[2]=TRUE)

)

## 8.2 FSPMM1

### 8.2.1 Primitive definitions

#### 8.2.1.1 Primitives exchanged between AP-Context and FSPMM1

Table 37 shows the service primitives including their associated parameters issued by the AP-Context and received by the FSPMM1.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 37 – Primitives issued by AP-Context to FSPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Init DP master Cl1.req | AP-Context | Bus Para | Refer to FAL Service Definition in IEC 61158-5-3 and in IEC 61158-3-3 |
| Reset DP master Cl1.req | AP-Context | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Load ARL DP master Cl1.req | AP-Context | Alarm Max,<br>List of ARL Entries | |
| Get ARL DP master Cl1.req | AP-Context | (none) | |
| Load CRL DP master Cl1.req | AP-Context | Update,<br>List of CRL Entries | |
| Get CRL DP master Cl1.req | AP-Context | (none) | |
| CRL Slave Activate.req | AP-Context | Activate | |
| CRL New Prm.req | AP-Context | New Prm | |
| CRL New Prm Data.req | AP-Context | Prm Data | |
| Abort DP master Cl1.req | AP-Context | AREP | |
| Mark DP master Cl1.req | AP-Context | AREP | |
| Set Mode DP master Cl1.req | AP-Context | AREP,<br>USIF State | |
| Load Bus Par DP master Cl1.req | AP-Context | Bus Para | |
| Delete SC DP master Cl1.req | AP-Context | Address | |
| Read Value DP master Cl1.req | AP-Context | Variable | |
| Get Slave Diag.req | AP-Context | AREP,<br>CREP | |
| Set Output.req | AP-Context | AREP,<br>CREP,<br>Slot Number,<br>Output Data,<br>Final | |
| Get Input.req | AP-Context | AREP,<br>CREP,<br>Slot Number | |
| Global Control.req | AP-Context | AREP,<br>Sync Command,<br>Freeze Command,<br>Group Select | |
| Read.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length | |
| Write.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Alarm Ack.req(+) | AP-Context | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Get Master Diag.rsp(+) | AP-Context | AREP,<br>Diagnosis Data | |
| Get Master Diag.rsp(-) | AP-Context | AREP,<br>Status | |
| Start Seq.rsp(+) | AP-Context | AREP,<br>Max Len Data Unit | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Start Seq.rsp(-) | AP-Context | AREP,<br>Status | |
| Download.rsp(+) | AP-Context | AREP | |
| Download.rsp(-) | AP-Context | AREP,<br>Status | |
| Upload.rsp(+) | AP-Context | AREP,<br>Data | |
| Upload.rsp(-) | AP-Context | AREP,<br>Status | |
| End Seq.rsp(+) | AP-Context | AREP | |
| End Seq.rsp(-) | AP-Context | AREP,<br>Status | |
| Act Param.rsp(+) | AP-Context | AREP | |
| Act Param.rsp(-) | AP-Context | AREP,<br>Status | |
| Set Time.req | AP-Context | AREP,<br>Time Value<br>Local Time Diff<br>Summertime<br>Accuracy<br>Synchronisation<br>Active<br>Announcement Hour | |
| Initiate Load.req | AP-Context | AREP,<br>Slot Number,<br>LR Index,<br>Load Type,<br>Load Image Size,<br>Intersegment<br>Request,<br>Timeout<br>User_Specific | |
| Pull Segment.req | AP-Context | AREP,<br>Slot Number,<br>LR Index,<br>Segment Length | |
| Push Segment.req | AP-Context | AREP,<br>Slot Number<br>LR Index,<br>Segment Length<br>Segment Number,<br>More Follows,<br>Data | |
| Terminate Load.req | AP-Context | AREP<br>Slot Number<br>LR Index | |
| Start.req | AP-Context | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Stop.req | AP-Context | AREP<br>Slot Number<br>FI Index | |
| Resume.req | AP-Context | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Reset.req | AP-Context | AREP<br>Slot Number<br>FI Index | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Call.req | FSPMS | AREP<br>Slot Number<br>Entity Number<br>FI Index<br>Execution Argument | |
| Get_FI_State.req | FSPMS | AREP<br>Slot Number<br>FI Index | |

Table 38 shows the service primitives including their associated parameters issued by the FSPMM1 and received by the AP-Context.

**Table 38 – Primitives issued by FSPMM1 to AP-Context**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Init DP master Cl1.cnf | FSPMM1 | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset DP master Cl1.cnf | FSPMM1 | (none) | |
| Load ARL DP master Cl1.cnf(+) | FSPMM1 | (none) | |
| Load ARL DP master Cl1.cnf(-) | FSPMM1 | Status | |
| Get ARL DP master Cl1.cnf(+) | FSPMM1 | Alarm Max,<br>List of ARL Entries | |
| Get ARL DP master Cl1.cnf(-) | FSPMM1 | Status | |
| Load CRL DP master Cl1.cnf(+) | FSPMM1 | (none) | |
| Load CRL DP master Cl1.cnf(-) | FSPMM1 | Status | |
| Get CRL DP master Cl1.cnf(+) | FSPMM1 | List of CRL Entries | |
| Get CRL DP master Cl1.cnf(-) | FSPMM1 | Status | |
| CRL Slave Activate.cnf(+) | FSPMM1 | | |
| CRL Slave Activate.cnf(-) | FSPMM1 | Status | |
| CRL New Prm.cnf(+) | FSPMM1 | | |
| CRL New Prm.cnf(-) | FSPMM1 | Status | |
| CRL New Prm Data.cnf(+) | FSPMM1 | | |
| CRL New Prm Data.cnf(-) | FSPMM1 | Status | |
| Mark DP master Cl1.cnf(+) | FSPMM1 | AREP,<br>Dia | |
| Mark DP master Cl1.cnf(-) | FSPMM1 | AREP,<br>Status | |
| Set Mode DP master Cl1.cnf(+) | FSPMM1 | AREP,<br>Bus Accessible | |
| Set Mode DP master Cl1.cnf(-) | FSPMM1 | AREP,<br>Bus Accessible | |
| Load Bus Par DP master Cl1.cnf | FSPMM1 | Status | |
| Delete SC DP master Cl1.cnf | FSPMM1 | Status | |
| Read Value DP master Cl1.cnf | FSPMM1 | Value, Status | |
| Get Slave Diag.cnf(+) | FSPMM1 | AREP,<br>CREP,<br>Diag Data | |
| Get Slave Diag.cnf(-) | FSPMM1 | AREP,<br>CREP | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Set Output.cnf(+) | FSPMM1 | AREP,<br>CREP,<br>Slot Number | |
| Set Output.cnf(-) | FSPMM1 | AREP,<br>CREP,<br>Slot Number,<br>Status | |
| Get Input.cnf(+) | FSPMM1 | AREP,<br>CREP,<br>Slot Number,<br>Input Data | |
| Get Input.cnf(-) | FSPMM1 | AREP,<br>CREP,<br>Slot Number,<br>Status | |
| Global Control.cnf(+) | FSPMM1 | AREP | |
| Global Control.cnf(-) | FSPMM1 | AREP,<br>Status | |
| Read.cnf(+) | FSPMM1 | AREP,<br>Length,<br>Data | |
| Read.cnf(-) | FSPMM1 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.cnf(+) | FSPMM1 | AREP,<br>Length | |
| Write.cnf(-) | FSPMM1 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Alarm Ack.cnf(+) | FSPMM1 | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Alarm Ack.cnf(-) | FSPMM1 | AREP<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Get Master Diag.ind | FSPMM1 | AREP,<br>Mdiag Identifier | |
| Start Seq.ind | FSPMM1 | AREP,<br>Area Code,<br>Timeout | |
| Download.ind | FSPMM1 | AREP,<br>Area Code,<br>Add Offset,<br>Data | |
| Upload.ind | FSPMM1 | AREP,<br>Area Code,<br>Add Offset,<br>Data Len | |
| End Seq.ind | FSPMM1 | AREP | |
| Act Param.ind | FSPMM1 | AREP,<br>Area Code,<br>Activate | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Act Para Brct.ind | FSPMM1 | AREP,<br>Area Code | |
| New Slave Diag.ind | FSPMM1 | AREP,<br>CREP | |
| New Input.ind | FSPMM1 | AREP | |
| Alarm Notification.ind | FSPMM1 | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr,<br>Add Ack,<br>Alarm Specifier,<br>Alarm Data | |
| DP master Cl1 Mode Changed.ind | FSPMM1 | AREP,<br>USIF State | |
| DP master Cl1 Event.ind | | Event,<br>Add Info | |
| DP master Cl1 Started.ind | FSPMM1 | AREP | |
| DP master Cl1 Stopped.ind | FSPMM1 | AREP | |
| Abort.ind | FSPMM1 | AREP | |
| DP master Cl1 Reject.ind | FSPMM1 | AREP,<br>Reason | |
| DP master Cl1 Fault.ind | FSPMM1 | ( none ) | |
| SYNCH.ind | FSPMM1 | AREP | |
| SYNCH Delayed.ind | FSPMM1 | AREP,<br>$T_{SH}$ | |
| DX Finished.ind | FSPMM1 | AREP | |
| Set Time.ind | FSPMM1 | AREP,<br>Time Value,<br>Local Time Diff,<br>Summertime,<br>Accuracy,<br>Synchronisation Active,<br>Announcement Hour | |
| Set Time.cnf | FSPMM1 | AREP,<br>Status | |
| Sync Interval Violation.ind | FSPMM1 | AREP | |
| Initiate Load.cnf(+) | FSPMM1 | AREP,<br>Actual LR Size,<br>Max Response Delay,<br>Max Segment Length<br>User Specific | |
| Initiate Load.cnf(-) | FSPMM1 | AREP,<br>Error Code | |
| Pull Segment.cnf(+) | FSPMM1 | AREP,<br>Segment Length,<br>Segment Number,<br>More Follows,<br>Data | |
| Pull Segment.cnf(-) | FSPMM1 | AREP,<br>Error Code | |
| Push Segment.cnf(+) | FSPMM1 | AREP | |
| Push Segment.cnf(-) | FSPMM1 | AREP,<br>Error Code | |
| Terminate Load.cnf(+) | FSPMM1 | AREP | |
| Terminate Load.cnf(-) | FSPMM1 | AREP,<br>Error Code | |
| Start.cnf(+) | FSPMM1 | AREP | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Start.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Stop.cnf(+) | FSPMM1 | AREP | |
| Stop.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Resume.cnf(+) | FSPMM1 | AREP | |
| Resume.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Reset.cnf(+) | FSPMM1 | AREP | |
| Reset.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Call.cnf(+) | FSPMM1 | AREP<br>Result Argument | |
| Call.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Get FI State.cnf(+) | FSPMM1 | AREP<br>FI State | |
| Get FI State.cnf(-) | FSPMM1 | AREP<br>Error Code | |

#### 8.2.1.2 Parameters of FSPMM1 primitives

The parameters used with the primitives exchanged between the FSPMM1 and the AP-Context are described in FAL Service Definition in IEC 61158-5-3.

### 8.2.2 State machine description

This Machine is used to co-ordinate the Interactions between AP-Context and DMPMM1, MSCY1M, MSAC1M, MSAL1M and MMAC1. As the support of several State Machines for the communication with individual slave as well as timing constrains of the operation of slave require a consistent scheduling, this task is accomplished by FSPMM1 a s well.

For each DP-slave an individual set of state machines (MSCY1M, MSAL1M, MSAC1M) will be established which will be controlled by the Scheduler in the DP-master (Class 1). The Scheduler will be controlled by the AP-Context with special services. The Slave-Handlers(MSCY1M) are triggered sequentially from the FSPMS-Scheduler to perform the necessary actions. The FSPMM1 notifies the operation modes CLEAR and OPERATE in fixed time intervals (Dx_Control_Timer) to the DP-slaves. This occurs also at state transitions of FSPMM1.

The Scheduler provides four operation modes for the User:

OFFLINE

STOP

CLEAR

OPERATE

**Local Variables**

**Go_AClr**
(Boolean)

State variable which indicates by the value True that after finishing the current Slave cycle the FSPMM1 has to force all DP-slaves into a safe state (Scheduler-State = CLEAR).

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Internal_Go_AClr**
(Boolean)

Only after finishing the current Slave cycle the FSPMS has to go to state CLEAR if Go_AClr is True. During each cycle the state variable Internal_Go_AClr stores the actual value of Go_AClr. At the end of each cycle the value of Internal_Go_AClr is copied into Go_AClr.

**Mark_Active**
(Unsigned8)

Indicates the state of the local function Mark.

Range:

Possible values are:

0 => Mark is not active

1 => Mark is active, but not yet in execution

2 => Mark is active and in execution

**Diag_Active**
(Boolean)

Used to store the information if at least one of the DP-slaves is not in the data exchange mode (True). The information is issued to the User with Mark.cnf.

**UGC_Count**
(Unsigned8)

Global_Control requests of the User are possible at any time during the FSPMM1 operation modes CLEAR and OPERATE. Confirmations of these requests will be passed to the User. The FSPMM1 uses the service Global_Control to send cyclic Global_Control requests with its actual operation mode to all assigned DP-slaves. Confirmations of cyclic Global_Control requests have to be utilised by the FSPMM1. The FSPMM1 has to know which Global_Control.cnf belongs to which previously called Global_Control.req It is not possible to resolve this relationship only by using DL services. Therefore the FSPMM1 shall get this relationship information by using the sequential order of the received Global_Control confirmations. The counters UGC_Count and CGC_Count are used to store information about the order of the requests and also information about the order of the confirmations.

With the UGC_Count counter all User Global_Control requests are counted incrementally. The initial value is 0. It is decremented with each Global_Control.cnf. In addition the counter is related with the parameter Bus_Para.Max_User_Global_Control.

Range: 0 .. Bus_Para.Max_User_Global_Control

**CGC_Count**
(Unsigned8)

With this counter all cyclic Global_Control requests are counted which where started by the Scheduler. The initial value is 0. With each new cyclic Global_Control.req UGC_Count is incremented and its value is copied to CGC_Count. UGC_Count also is decremented with each new Global_Control.cnf. If this counter reaches the value 1, the actual Global_Control.cnf is a confirmation for a cyclic Global_Control.req

Range: 0 .. Bus_Para.Max_User_Global_Control

**Bus_Accessible**
(Boolean)

Indicates the actual state of the bus system. If services can be sent, its value is True.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Act_Rem_Add**
(Unsigned8)

This variable stores the actually used station address of a DP-slave.

Range: 0 .. 125

**Act_msac1m_Max_L_sdu_len**
(Unsigned8)

This variable is used to store the actual value of the parameter msac1m_Max_L_sdu_len.

Range: 0 .. 240

**User_Sched_Reset**
(Boolean)

This variable indicates whether the actual reset was started by the User (AP-Context) or by any other State Machine.

**Act_New_Bus_Para**
(Bus_Para)

This structure is used to save an actual bus parameter set passed from the AP-Context to FSPMM1 temporarily.

**DX_Lock**
(Boolean)

This variable marks an active data exchange cycle to lock the data buffer and block user access. The value TRUE is set when the first Continue Slave Handler.req is issued until DX Finished.ind. The value remains FALSE until the next SYNCH.ind starts again the data exchange cycle.

**DX_Synch**
(Boolean)

This variable marks the interval between a SYNCH.ind and the next data exchange cycle. The value TRUE is set when the SYNCH.ind is received until the first Continue Slave Handler.req is issued. The value remains FALSE until the next SYNCH.ind.

**Chg Buffer**
(Boolean)

The value TRUE indicates that the buffers for input and output data below shall be changed with the next DX Finished.ind. It remains FALSE until the next Set Output.req (Final=TRUE) is received.

**Bfr_Set_Output**
(Octet String)

This variable is a local buffer for output data that belongs to the application. It allows asynchronous access from the application process to the output data in Buffered Synchronised Mode. The contents of the buffer is refreshed by the FSPMM1 when the application process has issued a Set Output.req (Final = TRUE) and the DX Finished.ind has been received. The buffer may accessed again after the next SYNCH.ind.

**Bfr_Get_Input**
(Octet String)

This variable is a local buffer for input data that belongs to the application. It allows asynchronous access from the application process to the input data in Buffered Synchronised Mode. The refreshment is done together with the Bfr_Set_Output.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Set_Output**
(Octet String)

This variable is a local buffer for output data that belongs to the network. It allows asynchronous access from the application process to the output data in Buffered Synchronised Mode. The contents of the buffer is refreshed by the FSPMM1 when the application process has issued a Set Output.req (Final = TRUE) and the DX Finished.ind has been received. The buffer may accessed again after the next SYNCH.ind.

**Get_Input**
(Octet String)

This variable is a local buffer for input data that belongs to the network. It allows asynchronous access from the application process to the input data in Buffered Synchronised Mode. The refreshment is done together with the Bfr_Set_Output

**LR_State**
(Array of Unsigned8)

This variable stores the state of the Load Region sequence of the corresponding CREP.

**CurrentExtendedFunctionNum**
(Array of Unsigned 8)

This variable stores the extended function number of the currently processed LR service of the corresponding CREP.

**CurrentSlotNumber**
(Array of Unsigned8)

This variable stores the slot number of the currently processed LR service of the corresponding CREP.

**Timers**

**Min_SI_Interval_Timer**
This timer supervises the minimal access control time of the DP-slaves. It is evaluated only within the FSPMM1 states CCLEAR and COPERATE.

**Dx_Control_Interval_Timer**
If this timer expires the broadcast service Global_Control is executed in order to inform the DP-slave whether the DP-master (Class 1) is in the state CLEAR or OPERATE. If the timer expires and the confirmation of the last Global_Control has not yet been received a physical bus access problem has occurred.

**Macros**

**INDICATE_BUS_ACCESSIBILITY**
If (Accessible = True) Bus_Accessible = True

### 8.2.3 FSPMM1 state table

Table 39 contains the complete description of the FSPMM1 state machine.

The following definition for a state set is used in Table 39.

t_state = STOP or CLEAR or OPERATE

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 39 – FSPMM1 state table**

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 1 | POWER-ON | Load ARL DP Master Cl1.req (Alarm Max, List of ARL Entries)<br>/valid parameters<br>=><br>Load ARL DP Master Cl1.cnf(+) | POWER-ON |
| 2 | POWER-ON | Load ARL DP Master Cl1.req (Alarm Max, List of ARL Entries)<br>/invalid parameters<br>=><br>Status := NO<br>Load ARL DP Master Cl1.cnf(-) (Status) | POWER-ON |
| 3 | POWER-ON | Get ARL DP Master Cl1.req<br>/ARL loaded<br>=><br>Get ARL DP Master Cl1.cnf(+) ( Alarm Max, List of ARL Entries ) | POWER-ON |
| 4 | POWER-ON | Get ARL DP Master Cl1.req<br>/ARL not loaded<br>=><br>Status := NO<br>Get ARL DP Master Cl1.cnf(-) (Status) | POWER-ON |
| 5 | POWER-ON | Load CRL DP Master Cl1.req ( List of CRL Entries )<br>/valid parameters<br>=><br>Load CRL DP Master Cl1.cnf(+) | POWER-ON |
| 6 | POWER-ON | Load CRL DP Master Cl1.req ( List of CRL Entries )<br>/invalid parameters<br>=><br>Status := NO<br>Load CRL DP Master Cl1.cnf(-) (Status) | POWER-ON |
| 7 | POWER-ON | Get CRL DP Master Cl1.req<br>/CRL loaded<br>=><br>Get CRL DP Master Cl1.cnf(+) (List of CRL Entries) | POWER-ON |
| 8 | POWER-ON | Get CRL DP Master Cl1.req<br>/CRL not loaded<br>=><br>Status := NO<br>Get CRL DP Master Cl1.cnf(-) (Status) | POWER-ON |
| 9 | POWER-ON | FSPMM1_Init DP Master Cl1.req(Bus_Para)<br>=><br>User_Init:=TRUE<br>USER_ChangedCritical_Para := FALSE<br>DX_Lock := FALSE<br>Chg_Buffer := FALSE<br>Current Extended Function Num[CREP]:=No_Service<br>LR_State[CREP]:=W-LR-REQ<br>DMPMM1_Minit_DLL.req(Bus_Para) | INIT-DMPM |
| 10 | INIT-DMPM | DMPMM1_Minit_DLL.cnf<br>=><br>Act_Rem_Add := 0<br>MSAC1M_MInit_MSAC1.req(Rem_Add:=Act_Rem_Add) | INIT-AC1 |
| 11 | — | — | — |
| 12 | INIT-AC1 | MSAC1M_MInit_MSAC1.cnf(Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add := Act_Rem_Add + 1<br>MSAC1M_MInit_MSAC1.req(Rem_Add:=Act_Rem_Add) | INIT-AC1 |
| 13 | INIT-AC1 | MSAC1M_MInit_MSAC1.cnf(Rem_Add)<br>/Act_Rem_Add=125<br>=><br>Act_Rem_Add := 0<br>MSAL1M_MInit.req(Rem_Add:=Act_Rem_Add) | INIT-AL1 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 14 | INIT-AL1 | MSAL1M_MInit.cnf(Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add := Act_Rem_Add + 1<br>MSAL1M_MInit.req(Rem_Add:=Act_Rem_Add) | INIT-AL1 |
| 15 | INIT-AL1 | MSAL1M_MInit.cnf(Rem_Add)<br>/Act_Rem_Add = 125<br>=><br>Act_Rem_Add := 0<br>MSCY1M_Minit_MS0.req(Rem_Add:=Act_Rem_Add) | INIT-CY1 |
| 16 | INIT-CY1 | MSCY1M_Minit_MS0.cnf(Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add := Act_Rem_Add + 1<br>MSCY1M_Minit_MS0.req(Rem_Add:=Act_Rem_Add) | INIT-CY1 |
| 17 | INIT-CY1 | MSCY1M_Minit_MS0.cnf(Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MM)=TRUE<br>=><br>MMAC1_Minit_MM.req | INIT-MM1 |
| 18 | INIT-CY1 | MSCY1M_Minit_MS0.cnf(Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MM)=FALSE &&<br>USER_ChangedCritical_Para = FALSE<br>=><br>Bus_accessible := FALSE<br>if(USER_SetMode_Offline=FALSE)<br>FSPMM1_Init DP Master CI1.cnf<br>else<br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible)<br>USER_SetMode_Offline:=FALSE<br>endif | OFFLINE |
| 19 | INIT-CY1 | MSCY1M_Minit_MS0.cnf(Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MM)=FALSE &&<br>USER_ChangedCritical_Para = TRUE<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para:=Act_New_Bus_Para) | LDBP |
| 20 | INIT-MM1 | MMAC1_Minit_MM.cnf<br>/USER_ChangedCritical_Para = TRUE<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para:=Act_New_Bus_Para) | LDBP |
| 21 | INIT-MM1 | MMAC1_Minit_MM.cnf<br>/USER_ChangedCritical_Para = FALSE<br>=><br>Bus_accessible := FALSE<br>if(USER_SetMode_Offline=FALSE)<br>FSPMM1_Init DP Master CI1.cnf<br>else<br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible)<br>USER_SetMode_Offline:=FALSE<br>endif | OFFLINE |
| 22 | OFFLINE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State <> Stop && USIF_State <> Offline<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(-)(AREP, Bus_accessible) | OFFLINE |
| 23 | OFFLINE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Offline<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible) | OFFLINE |
| 24 | OFFLINE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop && Bus_Para invalid<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(-)(AREP, Bus_accessible) | OFFLINE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 25 | OFFLINE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop && Bus_Para valid<br>=><br>Mark_Active := 0<br>Diag_Active := False<br>DMPMM1_Set_Bus_Par.req(Bus_Para) | LDBP |
| 26 | OFFLINE | FSPMM1_Mark.req(AREP)<br>=><br>FSPMM1_Mark.cnf(-)(AREP, Status:=NO) | OFFLINE |
| 27 | OFFLINE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | OFFLINE |
| 28 | OFFLINE | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=NO) | OFFLINE |
| 29 | OFFLINE | FSPMM1_Delete_SC DP Master CI1.req(Address)<br>=><br>DMPMM1_Delete_SC.req(Address) | DELSC-OFF |
| 30 | OFFLINE | FSPMM1_Read_Value DP Master CI1.req(Variable)<br>=><br>DMPMM1_Read_Value.req(Variable) | RDVAL-OFF |
| 31 | LDBP | DMPMM1_Set_Bus_Par.cnf<br>/USER_ChangeCritical_Para = FALSE<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible) | STOP |
| 32 | LDBP | DMPMM1_Set_Bus_Par.cnf<br>/USER_ChangeCritical_Para = TRUE<br>=><br>FSPMM1_Load_Bus_Para.cnf(+) | STOP |
| 33 | DELSC-OFF | DMPMM1_Delete_SC.cnf<br>=><br>FSPMM1_Delete_SC DP Master CI1.cnf | OFFLINE |
| 34 | RDVAL-OFF | DMPMM1_Read_Value.cnf(Status,Value)<br>=><br>FSPMM1_Read_Value DP Master CI1.cnf(Status, Value) | OFFLINE |
| 35 | STOP | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Offline<br>=><br>USER_SetMode_Offline:=TRUE | RESET |
| 36 | STOP | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible) | STOP |
| 37 | STOP | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(-)(AREP, Bus_accessible) | STOP |
| 38 | STOP | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Clear<br>=><br>UGC_Count := 0<br>CGC_Count := 0<br>Bus_Acessible := True<br>Go_AClr := Bus_Para.Error_Action_Flag<br>Internal_Go_AClr := False<br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>MSCY1M_Start_Slave_Handler.req(0 .. 125) | CLEAR |
| 39 | STOP | FSPMM1_Mark.req(AREP)<br>=><br>FSPMM1_Mark.cnf(-)(AREP, Status:=NO) | STOP |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 40 | STOP | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | STOP |
| 41 | STOP | FSPMM1_Load_Bus_Par DP Master Cl1.req(Bus_Para)<br>/Bus_Para invalid<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=IV) | STOP |
| 42 | STOP | FSPMM1_Load_Bus_Par DP Master Cl1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters unchanged)<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para) | LDBP-ST-CPU |
| 43 | STOP | FSPMM1_Load_Bus_Par DP Master Cl1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters changed)<br>=><br>Act_New_Bus_Para := Bus_Para<br>User_ChangedCritical_Para := TRUE | RESET |
| 44 | STOP | FSPMM1_Delete_SC DP Master Cl1.req(Address)<br>=><br>DMPMM1_Delete_SC.req(Address) | DELSC-ST |
| 45 | STOP | FSPMM1_Read_Value DP Master Cl1.req(Variable)<br>=><br>DMPMM1_Read_Value.req(Variable) | RDVAL-ST |
| 46 | LDBP-ST-CPU | DMPMM1_Set_Bus_Par.cnf<br>=><br>FSPMM1_Load_Bus_Para.cnf(+) | STOP |
| 47 | DELSC-ST | DMPMM1_Delete_SC.cnf<br>=><br>FSPMM1_Delete_SC DP Master Cl1.cnf | STOP |
| 48 | RDVAL-ST | DMPMM1_Read_Value.cnf(Status,Value)<br>=><br>FSPMM1_Read_Value DP Master Cl1.cnf(Status, Value) | STOP |
| 49 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Offline<br>=><br>FSPMM1_Set_Mode DP Master Cl1.cnf(-)(AREP, Bus_accessible) | CLEAR |
| 50 | CLEAR | FSPMM1_Mark DP Master Cl1.req(AREP)<br>/Mark_Active = 0<br>=><br>Mark_Active := 1 | CLEAR |
| 51 | CLEAR | FSPMM1_Mark DP Master Cl1.req(AREP)<br>/Mark_Active <> 0<br>=><br>FSPMM1_Mark.cnf(-)(AREP, Status:=NO) | CLEAR |
| 52 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop && Mark_Active = 0<br>=><br>Stop Dx_Control_Interval_Timer<br>MSCY1M_Stop_Slave_Handler.req(0..125) | SLHSTP |
| 53 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop && Mark_Active <> 0<br>=><br>Mark_Active := 0<br>Stop Dx_Control_Interval_Timer<br>FSPMM1_Mark.cnf(-)(AREP; Status:=NO)<br>MSCY1M_Stop_Slave_Handler.req(0..125) | SLHSTP |
| 54 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Clear<br>=><br>FSPMM1_Set_Mode DP Master Cl1.cnf(+)(AREP, Bus_accessible) | CLEAR |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 55 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate &&<br>Go_AClr = False && Internal_Go_AClr = False &&<br>CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2 | SGC-CO-WGC |
| 56 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate &&<br>Go_AClr = False && Internal_Go_AClr = False &&<br>CGC_Count = 0<br>=><br>UGC_Count := UGC_Count+1<br>CGC_Count := UGC_Count<br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=0,<br>Group_Select:=0) | SGC-CO |
| 57 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate &&<br>(Go_AClr = True \|\| Internal_Go_AClr = True)<br>=><br>FSPMM1_Set_Mode DP Master Cl1.cnf(-)(AREP, Bus_accessible) | CLEAR |
| 58 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/NOT received all MSCY1M_Start_Slave_Handler.cnf<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | CLEAR |
| 59 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count = 0 && UGC_Count >=<br>Bus_Para.Max_User_Global_Control<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | CLEAR |
| 60 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count = 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control &&<br>(Isochronous Mode = Not Synchronized \|\|<br>Group_Select < 0x80)<br>=><br>UGC_Count := UGC_Count+1<br>DMPMM1_Global_Control.req(Rem_Add:=CREP.Rem_Add,<br>Control_Command.1:=1,<br>Group_Select) | CLEAR |
| 61 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count > 0 && UGC_Count >=<br>Bus_Para.Max_User_Global_Control+1<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | CLEAR |
| 62 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count > 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control+1 &&<br>(Isochronous Mode = Not Synchronized \|\|<br>Group_Select < 0x80)<br>=><br>UGC_Count := UGC_Count+1<br>DMPMM1_Global_Control.req(Rem_Add:=CREP.Rem_Add,<br>Control_Command.1:=1,<br>Group_Select) | CLEAR |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 63 | CLEAR | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 0<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | CLEAR |
| 64 | CLEAR | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := 0 | CLEAR |
| 65 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count = 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control &&<br>Isochronous Mode = (Buffered Synchronized \|\| Enhanced Synchronized) &&<br>Group_Select >= 0x80<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=GE) | CLEAR |
| 66 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count > 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control+1 &&<br>Isochronous Mode = (Buffered Synchronized \|\| Enhanced Synchronized) &&<br>Group_Select >= 0x80<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=GE) | CLEAR |
| 67 | CLEAR | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | CLEAR |
| 68 | CLEAR | MSCY1M_Start_Slave_Handler.cnf(Rem_Add)<br>/NOT received all MSCY1M_Start_Slave_Handler.cnf<br>=><br>ignore | CLEAR |
| 69 | CLEAR | MSCY1M_Start_Slave_Handler.cnf(Rem_Add)<br>/received all MSCY1M_Start_Slave_Handler.cnf && Isochronous Mode = Not Synchronized<br>=><br>Internal_Go_AClr := False<br>Start Min_SI_Interval_Timer(Bus_Para.Min_Slave_Interval)<br>Set_Mode.cnf(Status:=OK, Bus_Accessible)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |
| 70 | CLEAR | MSCY1M_Start_Slave_Handler.cnf(Rem_Add)<br>/received all MSCY1M_Start_Slave_Handler.cnf && Isochronous Mode <> Not Synchronized<br>=><br>Internal_Go_AClr := False<br>Set_Mode.cnf(Status:=OK, Bus_Accessible)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |
| 71 | CLEAR | DMPMM1_SYNCH.ind<br>=><br>FSPMM1_DP Master Cl1 Fault.ind | RESET |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 72 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/NOT received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active = 2<br>=><br>Diag_Active := Diag_Active OR Diag<br>Internal_Go_AClr := Internal_Go_AClr OR<br>(Bus_Para.Error_Action_Flag<br>AND NOT No_AClr) | CLEAR |
| 73 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/NOT received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active <> 2<br>=><br>Internal_Go_AClr := Internal_Go_AClr OR<br>(Bus_Para.Error_Action_Flag<br>AND NOT No_AClr) | CLEAR |
| 74 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active = 0<br>=><br>Internal_Go_AClr := Internal_Go_AClr OR (Bus_Para.Error_Action_Flag AND NOT No_AClr)<br>DX_Lock := FALSE | CCLEAR |
| 75 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active = 1<br>=><br>Mark_Active:=2<br>Diag_Active := False<br>Internal_Go_AClr := Internal_Go_AClr OR (Bus_Para.Error_Action_Flag AND NOT No_AClr)<br>DX_Lock := FALSE | CCLEAR |
| 76 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active = 2<br>=><br>Diag_Active := Diag_Active OR Diag<br>Internal_Go_AClr := Internal_Go_AClr OR (Bus_Para.Error_Action_Flag AND NOT No_AClr)<br>Mark_Active:=0<br>DX_Lock := FALSE<br>FSPMM1_Mark.cnf(+)(AREP, Dia:=Diag_Active) | CCLEAR |
| 77 | CLEAR | Dx_Control_Interval_Timer expired<br>/CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>UGC_Count := UGC_Count+1<br>CGC_Count := UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2,<br>Group_Select:=0) | CLEAR |
| 78 | CLEAR | Dx_Control_Interval_Timer expired<br>/CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible := False | CLEAR |
| 79 | CLEAR | FSPMM1_Load_Bus_Par DP Master Cl1.req(Bus_Para)<br>/Bus_Para invalid<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=IV) | CLEAR |
| 80 | CLEAR | FSPMM1_Load_Bus_Par DP Master Cl1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters changed)<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=NO) | CLEAR |
| 81 | CLEAR | FSPMM1_Load_Bus_Par DP Master Cl1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters unchanged)<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para) | LDBP-CL-CPU |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 82 | CLEAR | FSPMM1_Delete_SC DP Master Cl1.req(Address)<br>=><br>DMPMM1_Delete_SC.req(Address) | DELSC-CL |
| 83 | CLEAR | FSPMM1_Read_Value DP Master Cl1.req(Variable)<br>=><br>DMPMM1_Read_Value.req(Variable) | RDVAL-CL |
| 84 | SLHSTP | MSCY1M_Stop_Slave_Handler.cnf(Rem_Add)<br>/NOT received all MSCY1M_Stop_Slave_Handler.cnf<br>=><br>ignore | SLHSTP |
| 85 | SLHSTP | MSCY1M_Stop_Slave_Handler.cnf(Rem_Add)<br>/received all MSCY1M_Stop_Slave_Handler.cnf<br>=><br>Set_Mode.cnf(Status:=OK, Bus_Accessible) | STOP |
| 86 | SGC-CO-WGC | Dx_Control_Interval_Timer expired<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible := False<br>FSPMM1_Set_Mode DP Master Cl1.cnf(-)(AREP, Bus_accessible) | CLEAR |
| 87 | SGC-CO-WGC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | SGC-CO-WGC |
| 88 | SGC-CO-WGC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>CGC_Count := UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=0,<br>Group_Select:=0) | SGC-CO |
| 89 | SGC-CO | Dx_Control_Interval_Timer expired<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible := False<br>Set_Mode.cnf(Status:=OK, Bus_Accessible) | OPERATE |
| 90 | SGC-CO | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | SGC-CO |
| 91 | SGC-CO | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := 0<br>FSPMM1_Set_Mode DP Master Cl1.cnf(+)(AREP, Bus_accessible) | OPERATE |
| 92 | CCLEAR | Min_Sl_Interval_Timer expired<br>=><br>Go_AClr := Internal_Go_AClr<br>Internal_Go_AClr := False<br>Start Min_Sl_Interval_Timer(Bus_Para.Min_Slave_Interval)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 93 | CCLEAR | DMPMM1_SYNCH.ind<br>=><br>DX_Lock := TRUE<br>Output_Data = Set_Output<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=False)<br>MSCY1M_Set_Output.req(Rem_Add:=0..125, Slot_Number:=ALL, Output_Data)<br>MSCY1M_Get_Input.req(Rem_Add:=0..125, Slot_Number:=ALL)<br>FSPMM1_SYNCH.ind (AREP) | CLEAR |
| 94 | LDBP-CL-CPU | DMPMM1_Set_Bus_Par.cnf<br>=><br>FSPMM1_Load_Bus_Para.cnf(+) | CLEAR |
| 95 | DELSC-CL | DMPMM1_Delete_SC.cnf<br>=><br>FSPMM1_Delete_SC DP Master Cl1.cnf | CLEAR |
| 96 | RDVAL-CL | DMPMM1_Read_Value.cnf(Status,Value)<br>=><br>FSPMM1_Read_Value DP Master Cl1.cnf(Status, Value) | CLEAR |
| 97 | OPERATE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State <> Clear && USIF_State <> Operate<br>=><br>FSPMM1_Set_Mode DP Master Cl1.cnf(-)(AREP, Bus_accessible) | OPERATE |
| 98 | OPERATE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate<br>=><br>FSPMM1_Set_Mode DP Master Cl1.cnf(+)(AREP, Bus_accessible) | OPERATE |
| 99 | OPERATE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Clear && CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2) | SGC-OC-WGC |
| 100 | OPERATE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Clear && CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2<br>UGC_Count := UGC_Count+1<br>CGC_Count := UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2, Group_Select:=0) | SGC-OC |
| 101 | OPERATE | FSPMM1_Mark DP Master Cl1.req(AREP)<br>/Mark_Active = 0<br>=><br>Mark_Active := 1 | OPERATE |
| 102 | OPERATE | FSPMM1_Mark DP Master Cl1.req(AREP)<br>/Mark_Active <> 0<br>=><br>FSPMM1_Mark.cnf(-)(AREP, Status:=NO) | OPERATE |
| 103 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 1<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | OPERATE |
| 104 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count = 0 && UGC_Count >=<br>Bus_Para.Max_User_Global_Control<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | OPERATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 105 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count = 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control &&<br>(Isochronous Mode = Not Synchronized ||<br>Group_Select < 0x80)<br>=><br>UGC_Count := UGC_Count+1<br>DMPMM1_Global_Control.req(Rem_Add:=CREP.Rem_Add,<br>Control_Command.1:=0,<br>Group_Select) | OPERATE |
| 106 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count > 0 && UGC_Count >=<br>Bus_Para.Max_User_Global_Control+1<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | OPERATE |
| 107 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count > 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control+1 &&<br>(Isochronous Mode = Not Synchronized ||<br>Group_Select < 0x80)<br>=><br>UGC_Count := UGC_Count+1<br>DMPMM1_Global_Control.req(Rem_Add:=CREP.Rem_Add,<br>Control_Command.1:=0,<br>Group_Select) | OPERATE |
| 108 | OPERATE | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 0<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | OPERATE |
| 109 | OPERATE | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := 0 | OPERATE |
| 110 | OPERATE | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | OPERATE |
| 111 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count = 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control &&<br>Isochronous Mode = (Buffered Synchronized \|\| Enhanced Synchronized) &&<br>Group_Select >= 0x80<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=GE) | OPERATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 112 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count > 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control+1 &&<br>Isochronous Mode = (Buffered Synchronized \|\| Enhanced Synchronized) &&<br>Group_Select >= 0x80<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=GE) | OPERATE |
| 113 | OPERATE | DMPMM1_SYNCH.ind<br>=><br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 114 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/NOT received all MSCY1M_Cont_Slave_Handler.cnf &&<br>Mark_Active = 2<br>=><br>Diag_Active:=Diag_Active OR Diag<br>Go_AClr := Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr) | OPERATE |
| 115 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/NOT received all MSCY1M_Cont_Slave_Handler.cnf &&<br>Mark_Active <> 2<br>=><br>Go_AClr := Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr) | OPERATE |
| 116 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 0 &&<br>(Isochronous Mode = Not Synchronized \|\| Isochronous Mode = Enhanced Synchronized)<br>=><br>Go_AClr := Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>DX_Lock := FALSE<br>FSPMM1_DX_Finished.ind (AREP) | COPERATE |
| 117 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 1 &&<br>(Isochronous Mode = Not Synchronized \|\| Isochronous Mode = Enhanced Synchronized)<br>=><br>Mark_Active:=2<br>Diag_Active := False<br>Go_AClr := Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>DX_Lock := FALSE<br>FSPMM1_DX_Finished.ind (AREP) | COPERATE |
| 118 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 2 &&<br>(Isochronous Mode = Not Synchronized \|\| Isochronous Mode = Enhanced Synchronized)<br>=><br>Diag_Active:=Diag_Active OR Diag<br>Go_AClr := Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>Mark_Active:=0<br>DX_Lock := FALSE<br>FSPMM1_Mark.cnf(+)(AREP, Dia:=Diag_Active)<br>DX_Finished.ind (AREP) | COPERATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 119 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 0 &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Go_AClr := Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>DX_Lock := FALSE<br>Chg_Bfr (Bfr_Get_Input, Get_Input)<br>FSPMM1_DX_Finished.ind (AREP) | COPERATE |
| 120 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 1 &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Mark_Active:=2<br>Diag_Active := False<br>Go_AClr := Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>DX_Lock := FALSE<br>Chg_Bfr (Bfr_Get_Input, Get_Input)<br>FSPMM1_DX_Finished.ind (AREP) | COPERATE |
| 121 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 2 &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Diag_Active:=Diag_Active OR Diag<br>Go_AClr := Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>Mark_Active:=0<br>DX_Lock := FALSE<br>FSPMM1_Mark.cnf(+)(AREP, Dia:=Diag_Active)<br>DX_Finished.ind (AREP) | COPERATE |
| 122 | OPERATE | Dx_Control_Interval_Timer expired<br>/CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>UGC_Count := UGC_Count+1<br>CGC_Count := UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=0,<br>Group_Select:=0) | OPERATE |
| 123 | OPERATE | Dx_Control_Interval_Timer expired<br>/CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_accessible := False | OPERATE |
| 124 | OPERATE | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/Bus_Para invalid<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=IV) | OPERATE |
| 125 | OPERATE | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters changed)<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=NO) | OPERATE |
| 126 | OPERATE | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters unchanged)<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para) | LDBP-OP-CPU |
| 127 | OPERATE | FSPMM1_Delete_SC DP Master CI1.req(Address)<br>=><br>DMPMM1_Delete_SC.req(Address) | DELSC-OP |
| 128 | OPERATE | FSPMM1_Read_Value DP Master CI1.req(Variable)<br>=><br>DMPMM1_Read_Value.req(Variable) | RDVAL-OP |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 129 | SGC-OC-WGC | Dx_Control_Interval_Timer expired<br>/Go_AClr = False<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible := False<br>FSPMM1_Set_Mode DP Master Cl1.cnf(-)(AREP, Bus_accessible) | OPERATE |
| 130 | SGC-OC-WGC | Dx_Control_Interval_Timer expired<br>/Go_AClr = True<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible := False | SGC-OC-WGC |
| 131 | SGC-OC-WGC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | SGC-OC-WGC |
| 132 | SGC-OC-WGC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>CGC_Count := UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2,<br>Group_Select:=0) | SGC-OC |
| 133 | SGC-OC | Dx_Control_Interval_Timer expired<br>/Go_AClr = False<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible := False<br>Set_Mode.cnf(Status:=OK, Bus_Accessible) | CLEAR |
| 134 | SGC-OC | Dx_Control_Interval_Timer expired<br>/Go_AClr = True<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible := False<br>FSPMM1_DP Master Cl1 Mode_Changed.ind(AREP, USIF_State:=Clear)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |
| 135 | SGC-OC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | SGC-OC |
| 136 | SGC-OC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1 && Go_AClr = False<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := 0<br>FSPMM1_Set_Mode DP Master Cl1.cnf(+)(AREP, Bus_accessible) | CLEAR |
| 137 | SGC-OC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1 && Go_AClr = True<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count := UGC_Count-1<br>CGC_Count := 0<br>FSPMM1_DP Master Cl1 Mode_Changed.ind(AREP, USIF_State:=Clear)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |
| 138 | COPERATE | Min_Sl_Interval_Timer expired<br>/Go_AClr = False<br>=><br>Start Min_Sl_Interval_Timer (Bus_Para.Min_Slave_Interval)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=False) | OPERATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 139 | COPERATE | Min_SI_Interval_Timer expired<br>/Go_AClr = True && CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer (Bus_Para.Data_Control_Time/2)<br>Start Min_SI_Interval_Timer (Bus_Para.Min_Slave_Interval) | SGC-OC-WGC |
| 140 | COPERATE | Min_SI_Interval_Timer expired<br>/Go_AClr = True && CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer (Bus_Para.Data_Control_Time/2)<br>UGC_Count := UGC_Count+1<br>CGC_Count := UGC_Count<br>Start Min_SI_Interval_Timer (Bus_Para.Min_Slave_Interval)<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2, Group_Select:=0) | SGC-OC |
| 141 | COPERATE | DMPMM1_SYNCH.ind<br>/Go_AClr = False<br>=><br>DX_Lock := TRUE<br>Output_Data = Set_Output<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=False)<br>MSCY1M_Set_Output.req(Rem_Add:=0..125, Slot_Number:=ALL, Output_Data)<br>MSCY1M_Get_Input.req(Rem_Add:=0..125, Slot_Number:=ALL)<br>FSPMM1_SYNCH.ind (AREP) | COPERATE |
| 142 | COPERATE | DMPMM1_SYNCH.ind<br>/Go_AClr = True && CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer (Bus_Para.Data_Control_Time/2)<br>DX_Lock := TRUE<br>Output_Data = Set_Output<br>MSCY1M_Set_Output.req(Rem_Add:=0..125, Slot_Number:=ALL, Output_Data)<br>MSCY1M_Get_Input.req(Rem_Add:=0..125, Slot_Number:=ALL)<br>FSPMM1_SYNCH.ind (AREP) | COPERATE |
| 143 | COPERATE | DMPMM1_SYNCH.ind<br>/Go_AClr = True && CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer (Bus_Para.Data_Control_Time/2)<br>UGC_Count := UGC_Count+1<br>CGC_Count := UGC_Count<br>DX_Lock := TRUE<br>Output_Data = Set_Output<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2, Group_Select:=0)<br>MSCY1M_Set_Output.req(Rem_Add:=0..125, Slot_Number:=ALL, Output_Data)<br>MSCY1M_Get_Input.req(Rem_Add:=0..125, Slot_Number:=ALL)<br>FSPMM1_SYNCH.ind (AREP) | COPERATE |
| 144 | LDBP-OP-CPU | DMPMM1_Set_Bus_Par.cnf<br>=><br>FSPMM1_Load_Bus_Para.cnf(+) | OPERATE |
| 145 | DELSC-OP | DMPMM1_Delete_SC.cnf<br>=><br>FSPMM1_Delete_SC DP Master Cl1.cnf | OPERATE |
| 146 | RDVAL-OP | DMPMM1_Read_Value.cnf(Status,Value)<br>=><br>FSPMM1_Read_Value DP Master Cl1.cnf(Status, Value) | OPERATE |
| 147 | RESET | /spontanous<br>=><br>Act_Rem_Add := 0<br>MSCY1M_Reset.req(Act_Rem_Add) | RESET-MS0 |
| 148 | RESET-MS0 | MSCY1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add := Act_Rem_Add+1<br>MSCY1M_Reset.req(Act_Rem_Add) | RESET-MS0 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 149 | RESET-MS0 | MSCY1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MS1)=TRUE<br>=><br>Act_Rem_Add := 0<br>MSAC1M_Reset.req(Act_Rem_Add) | RESET-MS1 |
| 150 | RESET-MS0 | MSCY1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MS1)=FALSE && &&<br>IsARExistent(MM1)=TRUE<br>=><br>MMAC1_Reset.req | RESET-MM1 |
| 151 | RESET-MS0 | MSCY1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MS1)=FALSE && &&<br>IsARExistent(MM1)=FALSE<br>=><br>DMPMM1_Reset.req | RESET-DMPM |
| 152 | RESET-MS1 | MSAC1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add=125<br>=><br>Act_Rem_Add := 0<br>MSAL1M_Reset.req(Act_Rem_Add) | RESET-MS1 |
| 153 | RESET-MS1 | MSAL1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add := Act_Rem_Add + 1<br>MSAL1M_Reset.req(Act_Rem_Add) | RESET-MS1 |
| 154 | RESET-MS1 | MSAL1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add = 125 && IsARExistent(MM1)=TRUE<br>=><br>MMAC1_Reset.req | RESET-MM1 |
| 155 | RESET-MS1 | MSAL1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add = 125 && IsARExistent(MM1)=FALSE<br>=><br>DMPMM1_Reset.req | RESET-DMPM |
| 156 | RESET-MM1 | MMAC1_Reset.cnf<br>=><br>DMPMM1_Reset.req | RESET-DMPM |
| 157 | RESET-DMPM | DMPMM1_Reset.cnf<br>/USER_SetMode_Offline = TRUE \|\| USER_ChangedCritical_Para = TRUE | INIT-DMPM |
| 158 | RESET-DMPM | DMPMM1_Reset.cnf<br>/USER_SetMode_Offline = FALSE && USER_ChangedCritical_Para = FALSE<br>=><br>if(USER_sched_Reset = TRUE)<br>FSPMM1_Reset DP Master CI1.cnf<br>endif | POWER-ON |
| 159 | t state | FSPMM1_Abort DP Master CI1.req(AREP)<br>/AREP.AR_Type=MS0<br>=><br>Current Extended Function Num[CREP]:=No_Service<br>LR_State[CREP]:=W-LR-REQ<br>MSCY1M_Abort.req(Rem_Add:=CREP.Rem_Add) | SAME |
| 160 | t state | FSPMM1_Abort DP Master CI1.req(AREP)<br>/AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=No_Service<br>LR_State[CREP]:=W-LR-REQ<br>MSAC1M_Abort.req(Rem_Add:=CREP.Rem_Add) | SAME |
| 161 | t state | FSPMM1_Abort DP Master CI1.req(AREP)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Abort.req(Rem_Add:=CREP.Rem_Add) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 162 | t state | FSPMM1_Get_Slave_Diag.req(AREP, CREP)<br>/AREP.AR_Type=MS0<br>=><br>MSCY1M_Get_Slave_Diag.req(CREP.Rem_Add) | SAME |
| 163 | t state | MSCY1M_Get_Slave_Diag.cnf(+)(Rem_Add, Diag_Data)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Get_Slave_Diag.cnf(+)(AREP, CREP, Diag_Data) | SAME |
| 164 | t state | MSCY1M_Get_Slave_Diag.cnf(-)(Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Get_Slave_Diag.cnf(-)(AREP, CREP) | SAME |
| 165 | t state | MSCY1M_Set_Output.cnf(+)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = (Not Synchronized \|\| Enhanced Synchronized)<br>=><br>FSPMM1_Set_Output.cnf(+)(AREP, CREP, Slot_Number) | SAME |
| 166 | t state | MSCY1M_Set_Output.cnf(-)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = (Not Synchronized \|\| Enhanced Synchronized)<br>=><br>Status := SC<br>FSPMM1_Set_Output.cnf(-)(AREP, CREP, Slot_Number, Status) | SAME |
| 167 | t state | MSCY1M_Get_Input.cnf(+)(Rem_Add, Slot_Number, Input_Data)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = (Not Synchronized \|\| Enhanced Synchronized)<br>=><br>FSPMM1_Get_Input.cnf(+)(AREP, CREP, Slot_Number, Input_Data) | SAME |
| 168 | t state | MSCY1M_Get_Input.cnf(-)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = (Not Synchronized \|\| Enhanced Synchronized)<br>=><br>Status := SC<br>FSPMM1_Get_Input.cnf(-)(AREP, CREP, Slot_Number, Status) | SAME |
| 169 | t state | FSPMM1_Read.req(AREP, Slot_Number, Index, Length)<br>/AREP.AR_Type=MS1<br>=><br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length) | SAME |
| 170 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Read.cnf(+)(AREP, Length, Data) | SAME |
| 171 | t state | MSAC1M_Read.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Read.cnf(-)(AREP, Rem_Add, Error_Decode, Error_Code_1, Error_Code_2) | SAME |
| 172 | t state | FSPMM1_Write.req(AREP, Slot_Number, Index, Length, Data)<br>/AREP.AR_Type=MS1<br>=><br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data) | SAME |
| 173 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Write.cnf(+)(AREP, Length) | SAME |
| 174 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Write.cnf(-)(AREP, Rem_Add, Error_Decode, Error_Code_1, Error_Code_2) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 175 | t state | FSPMM1_Alarm_Ack.req(AREP, Slot_Number, Alarm_Type, Seq_Nr)<br>/AREP.AR_Type=MS1<br>=><br>MSAL1M_Alarm_Ack.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Alarm_Type, Seq_Nr) | SAME |
| 176 | t state | MSAL1M_Alarm_Ack.cnf(+)(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Alarm_Ack.cnf(+)(AREP, Slot_Number, Alarm_Type,Seq_Nr) | SAME |
| 177 | t state | MSAL1M_Alarm_Ack.cnf(-)(Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Alarm_Ack.cnf(-)(AREP, Slot Number, Alarm Type,Seq Nr) | SAME |
| 178 | t state | FSPMM1_Get_Master_Diag.rsp(+)(AREP, Diagnosis_Data)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Get_Master_Diag.rsp(Status:=OK, Diagnosis_Data) | SAME |
| 179 | t state | FSPMM1_Get_Master_Diag.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Get_Master_Diag.rsp(Status) | SAME |
| 180 | t state | MMAC1_Get_Master_Diag.ind(Identifier)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Get_Master_Diag.ind(AREP, Mdiag_Identifier:=Identifier) | SAME |
| 181 | t state | FSPMM1_Start_Seq.rsp(+)(AREP, Max_Len_Data_Unit)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Start_Seq.rsp(Status:=OK, Max_Length_Data_Unit) | SAME |
| 182 | t state | FSPMM1_Start_Seq.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Start_Seq.rsp(Status) | SAME |
| 183 | t state | MMAC1_Start_Seq.ind(Req_Add, Area_Code, Timeout)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Start_Seq.ind(AREP, Area_Code, Timeout) | SAME |
| 184 | t state | FSPMM1_Download.rsp(+)(AREP)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Download.rsp(Status:=OK) | SAME |
| 185 | t state | FSPMM1_Download.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Download.rsp(Status) | SAME |
| 186 | t state | MMAC1_Download.ind(Req_Add, Area_Code, Address_Offset, Data)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Download.ind(AREP, Area_Code, Add_Offset, Data) | SAME |
| 187 | t state | FSPMM1_Upload.rsp(+)(AREP, Data)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Upload.rsp(Status:=OK, Data) | SAME |
| 188 | t state | FSPMM1_Upload.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Upload.rsp(Status) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 189 | t state | MMAC1_Upload.ind(Req_Add, Area_Code, Address_Offset, Data_Length) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_Upload.ind(AREP, Area_Code, Add_Offset, Data_Len:=Data_Length) | SAME |
| 190 | t state | FSPMM1_End_Seq.rsp(+)(AREP) /AREP.AR_Type=MM1 => MMAC1_End_Seq.rsp(Status:=OK) | SAME |
| 191 | t state | FSPMM1_End_Seq.rsp(-)(AREP, Status) /AREP.AR_Type=MM1 => MMAC1_End_Seq.rsp(Status) | SAME |
| 192 | t state | MMAC1_End_Seq.ind(Req_Add) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_End_Seq.ind(AREP) | SAME |
| 193 | t state | FSPMM1_Act_Param.rsp(+)(AREP) /AREP.AR_Type=MM1 => MMAC1_Act_Param.rsp(Status:=OK) | SAME |
| 194 | t state | FSPMM1_Act_Param.rsp(-)(AREP, Status) /AREP.AR_Type=MM1 => MMAC1_Act_Param.rsp(Status) | SAME |
| 195 | t state | MMAC1_Act_Param.ind(Area_Code, Activate) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_Act_Param.ind(AREP, Area_Code, Activate) | SAME |
| 196 | t state | MMAC1_Act_Para_Brct.ind(Area_Code) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_Act_Para_Brct.ind(AREP, Area_Code) | SAME |
| 197 | t state | MSAC1M_Reject.ind(Rem_Add, Status) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_DP Master Cl1 Reject.ind(AREP, Reason:=Status) | SAME |
| 198 | t state | MSAL1M_Started.ind(Rem_Add, Alarm_Limit) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_DP Master Cl1 Started.ind(AREP, Alarm_Limit) | SAME |
| 199 | t state | MSAL1M_Stopped.ind(Rem_Add) /SetContext(Rem_Add, Service)=TRUE => Current Extended Function Num[CREP]:=No_Service LR_State[CREP]:=W-LR-REQ FSPMM1_DP Master Cl1 Stopped.ind(AREP) | SAME |
| 200 | t state | MSAL1M_Alarm_Notification.ind(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr, Alarm_Specifier, Add_Ack, Alarm_Data) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_Alarm_Notification.ind(AREP, Alarm_Type, Slot_Number, Seq_Nr, Alarm_Specifier, Add_Ack, Alarm_Data) | SAME |
| 201 | t state | MSCY1M_New_Slave_Diag.ind(Rem_Add) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_New_Slave_Diag.ind(AREP, CREP) | SAME |
| 202 | t state | MSCY1M_New_Input.ind(Rem_Add) /SetContext(Rem_Add, Service)=TRUE => FSPMM1_New_Input.ind(AREP, CREP) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 203 | t state | FSPMM1_Set_Output.req(AREP, CREP, Slot_Number, Output_Data, Final)<br>/AREP.AR_Type=MS0 &&<br>(DX_Lock \|\| Chg_Buffer)<br>=><br>Status := SE<br>FSPMM1_Set_Output.cnf(-)(AREP, CREP, Slot_Number, Status) | SAME |
| 204 | t state | FSPMM1_Get_Input.req(AREP, CREP, Slot_Number)<br>/AREP.AR_Type=MS0 &&<br>(DX_Lock \|\| Chg_Buffer)<br>=><br>Status := SE<br>FSPMM1_Get_Input.cnf(-)(AREP, CREP, Slot_Number, Status) | SAME |
| 205 | t state | FSPMM1_Set_Output.req(AREP, CREP, Slot_Number, Output_Data, Final)<br>/AREP.AR_Type=MS0 &&<br>!DX_Lock && !Chg_Buffer && Isochronous Mode <> Buffered Synchronized<br>=><br>MSCY1M_Set_Output.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Output_Data) | SAME |
| 206 | t state | FSPMM1_Get_Input.req(AREP, CREP, Slot_Number)<br>/AREP.AR_Type=MS0 &&<br>!DX_Lock && !Chg_Buffer && Isochronous Mode <> Buffered Synchronized<br>=><br>MSCY1M_Get_Input.req(Rem_Add:=CREP.Rem_Add, Slot_Number) | SAME |
| 207 | t state | FSPMM1_Set_Output.req(AREP, CREP, Slot_Number, Output_Data, Final)<br>/AREP.AR_Type=MS0 &&<br>!DX_Lock && !Chg_Buffer && Isochronous Mode = Buffered Synchronized<br>=><br>Bfr_Set_Output := Output_Data<br>Chg_Buffer := Final<br>FSPMM1_Set_Output.cnf(+)(AREP, CREP, Slot_Number) | SAME |
| 208 | t state | FSPMM1_Get_Input.req(AREP, CREP, Slot_Number)<br>/AREP.AR_Type=MS0 &&<br>!DX_Lock && !Chg_Buffer && Isochronous Mode = Buffered Synchronized<br>=><br>Input_Data := Bfr_Get_Input<br>FSPMM1_Get_Input.cnf(+)(AREP, CREP, Slot_Number, Input_Data) | SAME |
| 209 | t state | MSCY1M_Set_Output.cnf(+)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized<br>=> | SAME |
| 210 | t state | MSCY1M_Set_Output.cnf(-)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Set_Output = Nil | SAME |
| 211 | t state | MSCY1M_Get_Input.cnf(+)(Rem_Add, Slot_Number, Input_Data)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized &&<br>NOT received all MSCY1M_Get_Input.cnf<br>=><br>Get_Input = Input_Data | SAME |
| 212 | t state | MSCY1M_Get_Input.cnf(+)(Rem_Add, Slot_Number, Input_Data)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized&&<br>NOT received all MSCY1M_Get_Input.cnf<br>=><br>Chg_Buffer := FALSE<br>Get_Input = Input_Data | SAME |
| 213 | t state | MSCY1M_Get_Input.cnf(-)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Get_Input = Nil | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 214 | ANY-STATE | DMPMM1_Event.ind(Event, Add_Info)<br>=><br>FSPMM1_DP Master Cl1 Event.ind(Event, Add_Info) | SAME |
| 215 | ANY-STATE | DMPMM1_SYNCH_Delayed.ind (TSH)<br>=><br>FSPMM1_SYNCH_Delayed.ind (AREP, TSH) | SAME |
| 216 | ANY-STATE | DMPMM1_Fault.ind<br>=><br>USER_sched_Reset := False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 217 | ANY-STATE | MSCY1M_Fault.ind<br>=><br>USER_sched_Reset := False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 218 | ANY-STATE | MSAL1M_Fault.ind<br>=><br>USER_sched_Reset := False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 219 | ANY-STATE | MSAC1M_Fault.ind<br>=><br>USER_sched_Reset := False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 220 | ANY-STATE | MMAC1_Fault.ind<br>=><br>USER_sched_Reset := False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 221 | ANY-STATE | FSPMM1_Reset DP Master Cl1.req<br>=><br>USER_sched_Reset := True | RESET |
| 222 | ANY-STATE | DMPMM1_SYCL_Time_Event.cnf (Send Delay Time, Status)<br>=><br>FSPMM1_SYCL_Time_Event.cnf (AREP, Send Delay Time, Status) | SAME |
| 223 | ANY-STATE | FSPMM1_SYCL_Time_Event.req (AREP)<br>=><br>DMPMM1_SYCL_Time_Event.req | SAME |
| 224 | ANY-STATE | FSPMM1_SYCL_Clock_Value.req (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status)<br>=><br>DMPMM1_SYCL_Clock_Value.req (Clock Value Time Event, Clock Value previous TE, Clock Value Status) | SAME |
| 225 | ANY-STATE | DMPMM1_SYCL_Clock_Value.cnf (Status)<br>=><br>FSPMM1_SYCL_Clock_Value.cnf (AREP, Status) | SAME |
| 226 | ANY-STATE | DMPMM1_SYCL_Clock_Value.ind (Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg)<br>=><br>FSPMM1_SYCL_Clock_Value.ind (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 227 | t state | FSPMM1_Initiate_Load.req ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Initiate_Load<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=10+sizeof(User Specific)<br>Data.w.IL.Extended_Function_Num:=Initiate_Load<br>Data.w.IL.LR_Index:=LR Index<br>Data.w.IL.Load_Type:=Load Type<br>Data.w.IL.Load_Image_Size:=Load Image Size<br>Data.w.IL.User_Specific:=User Specific<br>Data.w.IL.Intersegment_Request_Timeout:=Intersegment Request Timeout<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 228 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Initiate_Load<br>=><br>Slot_Number:=Current Slot Number[CREP]<br>Index:=255<br>Length:=10<br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 229 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Initiate_Load<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Actual LR Size:=Data.r.IL.Actual_LR_Size<br>Max Response Delay:=Data.r.IL.Max_Response_Delay<br>Max Segment Length:=Data.r.IL.Max_Segment_Length<br>User Specific:=Data.r.IL.User_Specific<br>FSPMM1_Initiate_Load.cnf(+) ( AREP, Actual LR Size, Max Response Delay, Max Segment Length, User Specific )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 230 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Initiate_Load<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Initiate_Load.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 231 | t state | FSPMM1_Pull_Segment.req ( AREP, Slot Number, LR Index, Segment Length )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Pull<br>Current LR Index[CREP]:=LR Index<br>Index:=255<br>Length:=Segment Length<br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length)<br>LR_State[CREP]:=W-LR-CNF | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 232 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Pull<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Segment Number:=Data.PULL.Sequence Number<br>More Follows:=Data.PULL.Options.More Follows<br>Data:=Data.PULL.Region Data<br>Segment Length:=Length-6<br>FSPMM1_Pull_Segment.cnf(+) ( AREP, Segment Length, Segment Number, More Follows, Data )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 233 | t state | MSAC1M_Read.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Pull<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Pull_Segment.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 234 | t state | FSPMM1_Push_Segment.req ( AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Push<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=6+Segment Length<br>Data.PUSH.Extended_Function_Num:=Push<br>Data.PUSH.LR_Index:=LR Index<br>Data.PUSH.Segment_Number:=Segment Number<br>Data.PUSH.Option.More_Follows:=More Follows<br>Data.PUSH.Data:=Data<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 235 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Push<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Segment Number:=Data.PUSH.Sequence Number<br>More Follows:=Data.PUSH.Options.More Follows<br>Data:=Data.PUSH.Region Data<br>Segment Length:=Length-6<br>FSPMM1_Push_Segment.cnf(+) ( AREP)<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 236 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Push<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Push_Segment.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 237 | t state | FSPMM1_Terminate_Load.req ( AREP, Slot Number, LR Index )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Terminate_Load<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=6<br>Data.TL.Extended_Function_Num:=Terminate_Load<br>Data.TL.LR_Index:=LR Index<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 238 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Terminate_Load<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Terminate_Load.cnf(+) ( AREP)<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 239 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Terminate_Load<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Terminate_Load.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 240 | t state | FSPMM1_Call.req ( AREP, Slot Number, Entity Number, FI Index, Execution Argument)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Call<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.w.CALL.Extended_Function_Num:=Call<br>Data.w.CALL.Entity_Number:= Entity_Number<br>Data.w.CALL.FI_Index:=FI Index<br>Data.w.CALL.Execution_Argument:=Execution Argument<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 241 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Call<br>=><br>Slot_Number:=Current Slot Number[CREP]<br>Index:=255<br>Length:=CREP.Max_PDU_Length<br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 242 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Call<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Result Argument:=Data.r.CALL.Result_Argument<br>FSPMM1_Call.cnf(+) ( AREP, Result Argument )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 243 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Call<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Call.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 244 | t state | FSPMM1_Start.req ( AREP, Slot Number, FI Index, Execution Argument)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Start<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.FIS.Extended_Function_Num:=Start<br>Data.FIS.FI_Index:=FI Index<br>Data.FIS.Execution_Argument:=Execution Argument<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 245 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Start<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Start.cnf(+) ( AREP )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 246 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Start<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Start.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 247 | t state | FSPMM1_Stop.req ( AREP, Slot Number, FI Index)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Stop<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.FIS.Extended_Function_Num:=Stop<br>Data.FIS.FI_Index:=FI Index<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 248 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Stop<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Stop.cnf(+) ( AREP )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 249 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Stop<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Stop.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 250 | t state | FSPMM1_Resume.req ( AREP, Slot Number, FI Index, Execution Argument)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Resume<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.FIS.Extended_Function_Num:=Resume<br>Data.FIS.FI_Index:=FI Index<br>Data.FIS.Execution_Argument:=Execution Argument<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 251 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Resume<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Resume.cnf(+) ( AREP )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 252 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Resume<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Resume.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 253 | t state | FSPMM1_Reset.req ( AREP, Slot Number, FI Index)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Reset<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.FIS.Extended_Function_Num:=Reset<br>Data.FIS.FI_Index:=FI Index<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 254 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Reset<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Reset.cnf(+) ( AREP )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 255 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Reset<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Reset.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 256 | t state | FSPMM1_Get_FI_State.req ( AREP, Slot Number, FI Index )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Get_State<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.w.STATE.Extended_Function_Num:=Get_State<br>Data.w.STATE.FI_Index:=FI Index<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 257 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Get_State<br>=><br>Slot_Number:=Current Slot Number[CREP]<br>Index:=255<br>Length:=5<br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 258 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Get_State<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FI State:=Data.r.STATE.State<br>FSPMM1_Get_FI_State.cnf (+) ( AREP, FI State )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 259 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Get_State<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Get_State.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 8.2.4 Functions

Table 40 contains the functions used by the FSPMM1, their arguments and their descriptions.

**Table 40 – Functions used by the FSPMM1**

| Function name | Description |
|---|---|
| SetContext (Rem_Add, Service-Id) | This function checks if there exist an entry for the inputs Rem_Add resp. Res_SAP and Service in the ARL and related CRL of the DP-master (Class 1).<br>A) If there exists an entry then it returns TRUE and sets the local context according to the current CREP and AREP.<br>B) Otherwise it returns FALSE. |
| IsARexistent (AR) | This function checks if there exist an entry with ARtype=AR. |
| Chg_Bfr(Buffer1, Buffer 2) | This function exchanges Buffer 1 and Buffer 2. |

## 8.3 FSPMM2

### 8.3.1 Primitive definitions

#### 8.3.1.1 Primitives exchanged between AP-Context and FSPMM2

Table 41 shows the service primitives including their associated parameters issued by the AP-Context and received by the FSPMM2.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 41 – Primitives issued by AP-Context to FSPMM2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Init DP master CI2.req | AP-Context | Bus Para | Refer to FAL Service Definition in IEC 61158-5-3 and in IEC 61158-3-3 |
| Reset DP master CI2.req | AP-Context | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Abort.req | AP-Context | AREP<br>Subnet,<br>Instance,<br>Reason Code | |
| Read Slave Diag.req | AP-Context | AREP | |
| Read Output.req | AP-Context | AREP | |
| Read Input.req | AP-Context | AREP | |
| Get Cfg.req | AP-Context | AREP | |
| Set Slave Add.req | AP-Context | AREP,<br>New Slave Add,<br>Ident Number,<br>No Add Chg,<br>Rem Slave Data | |
| Initiate.req | AP-Context | AREP,<br>Send Timeout,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Read.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length | |
| Write.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Data Transport.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Get Master Diag.req | AP-Context | AREP,<br>MDiag Identifier | |
| Start Seq.req | AP-Context | AREP,<br>Area Code,<br>Timeout | |
| Download.req | AP-Context | AREP,<br>Area Code,<br>Add Offset,<br>Data | |
| Upload.req | AP-Context | AREP,<br>Area Code,<br>Add Offset,<br>Data Len | |
| End Seq.req | AP-Context | AREP | |
| Act Param.req | AP-Context | AREP,<br>Area Code,<br>Activate | |
| Act Para Brct.req | AP-Context | AREP,<br>Area Code | |
| Load ARL DP master CI2.req | AP-Context | List of ARL Entries | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Get ARL DP master CI2.req | AP-Context | (none) | |
| Load CRL DP master CI2.req | AP-Context | List of CRL Entries | |
| Get CRL DP master CI2.req | AP-Context | (none) | |
| Initiate Load.req | AP-Context | AREP,<br>Slot Number,<br>LR Index,<br>Load Type,<br>Load Image Size,<br>Intersegment Request,<br>Timeout | |
| Pull Segment.req | AP-Context | AREP,<br>Slot Number,<br>LR Index,<br>Segment Length | |
| Push Segment.req | AP-Context | AREP,<br>Slot Number<br>LR Index,<br>Segment Length<br>Segment Number,<br>More Follows,<br>Data | |
| Terminate Load.req | AP-Context | AREP<br>Slot Number<br>LR Index | |
| Start.req | AP-Context | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Stop.req | AP-Context | AREP<br>Slot Number<br>FI Index | |
| Resume.req | AP-Context | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Reset.req | AP-Context | AREP<br>Slot Number<br>FI Index | |
| Call.req | FSPMS | AREP<br>Slot Number<br>Entity Number<br>FI Index<br>Execution Argument | |

Table 42 shows the service primitives including their associated parameters issued by the AP-Context and received by the FSPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 42 – Primitives issued by FSPMM2 to AP-Context**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Init DP master CI2.cnf | FSPMM2 | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | FSPMM2 | (none) | |
| Read Slave Diag.cnf(+) | FSPMM2 | AREP,<br>Diag Data | |
| Read Slave Diag.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Get Cfg.cnf(+) | FSPMM2 | AREP,<br>Cfg Data | |
| Get Cfg.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Read Output.cnf(+) | FSPMM2 | AREP,<br>Output Data | |
| Read Output.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Read Input.cnf(+) | FSPMM2 | AREP,<br>Input Data | |
| Read Input.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Set Slave Add.cnf(+) | FSPMM2 | AREP | |
| Set Slave Add.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Initiate.cnf(+) | FSPMM2 | AREP,<br>Max Len Data Unit,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Initiate.cnf(-) | FSPMM2 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Read.cnf(+) | FSPMM2 | AREP,<br>Length,<br>Data | |
| Read.cnf(-) | FSPMM2 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.cnf(+) | FSPMM2 | AREP,<br>Length | |
| Write.cnf(-) | FSPMM2 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Data Transport.cnf(+) | FSPMM2 | AREP,<br>Length,<br>Data | |
| Data Transport.cnf(-) | FSPMM2 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Get Master Diag.cnf(+) | FSPMM2 | AREP,<br>Diagnosis Data | |
| Get Master Diag.cnf(-) | FSPMM2 | AREP,<br>Status | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Start Seq.cnf(+) | FSPMM2 | AREP,<br>Max Len Data Unit | |
| Start Seq.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Download.cnf(+) | FSPMM2 | AREP | |
| Download.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Upload.cnf(+) | FSPMM2 | AREP,<br>Data | |
| Upload.cnf(-) | FSPMM2 | AREP,<br>Status | |
| End Seq.cnf(+) | FSPMM2 | AREP | |
| End Seq.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Act Param.cnf(+) | FSPMM2 | AREP | |
| Act Param.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Act Para Brct.cnf(+) | FSPMM2 | AREP | |
| Act Para Brct.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Event.ind | FSPMM2 | Event,<br>Add Info | |
| DP master CI2 Reject.ind | FSPMM2 | AREP,<br>Reason | |
| Abort.ind | FSPMM2 | AREP,<br>Locally Generated,<br>Subnet,<br>Instance,<br>Reason_Code,<br>Additional_Detail | |
| DP master CI2 Fault.ind | FSPMM2 | (none) | |
| Load ARL DP master CI2.cnf(+) | FSPMM2 | (none) | |
| Load ARL DP master CI2.cnf(-) | FSPMM2 | Status | |
| Get ARL DP master CI2.cnf(+) | FSPMM2 | List of ARL Entries | |
| Get ARL DP master CI2.cnf(-) | FSPMM2 | Status | |
| Load CRL DP master CI2.cnf(+) | FSPMM2 | (none) | |
| Load CRL DP master CI2.cnf(-) | FSPMM2 | Status | |
| Get CRL DP master CI2.cnf(+) | FSPMM2 | List of CRL Entries | |
| Get CRL DP master CI2.cnf(-) | FSPMM2 | Status | |
| Initiate Load.cnf(+) | FSPMM2 | AREP,<br>Actual LR Size,<br>Max Response Delay,<br>Max Segment Length | |
| Initiate Load.cnf(-) | FSPMM2 | AREP,<br>Error Code | |
| Pull Segment.cnf(+) | FSPMM2 | AREP,<br>Segment Length,<br>Segment Number,<br>More Follows,<br>Data | |
| Pull Segment.cnf(-) | FSPMM2 | AREP,<br>Error Code | |
| Push Segment.cnf(+) | FSPMM2 | AREP | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Push Segment.cnf(-) | FSPMM2 | AREP,<br>Error Code | |
| Terminate Load.cnf(+) | FSPMM2 | AREP | |
| Terminate Load.cnf(-) | FSPMM2 | AREP,<br>Error Code | |
| Start.cnf(+) | FSPMM2 | AREP | |
| Start.cnf(-) | FSPMM2 | AREP<br>Error Code<br>Function Invocation State | |
| Stop.cnf(+) | FSPMM2 | AREP | |
| Stop.cnf(-) | FSPMM2 | AREP<br>Error Code<br>Function Invocation State | |
| Resume.cnf(+) | FSPMM2 | AREP | |
| Resume.cnf(-) | FSPMM2 | AREP<br>Error Code<br>Function Invocation State | |
| Reset.cnf(+) | FSPMM2 | AREP | |
| Reset.cnf(-) | FSPMM2 | AREP<br>Error Code | |
| Call.cnf(+) | FSPMM2 | AREP<br>Result Argument | |
| Call.cnf(-) | FSPMM2 | AREP<br>Result Argument<br>Error Code | |

#### 8.3.1.2 Parameters of FSPMM2 primitives

The parameters used with the primitives exchanged between the FSPMM2 and the AP-Context are described in FAL Service Definition in IEC 61158-5-3.

### 8.3.2 State machine description

This Machine is used to co-ordinate the Interactions between AP-Context and DMPMM1, MSCY1M, MSAC1M, MSAL1M and MMAC1. As the support of several State Machines for the communication with individual slave as well as timing constrains of the operation of slave require a consistent scheduling, this task is accomplished by FSPMM1 as well.

**Local Variables**

**LR_State**
(Array of Unsigned8)

This variable stores the state of the Load Region sequence of the corresponding AREP.

**CurrentExtendedFunctionNum**
(Array of Unsigned 8)

This variable stores the extended function number of the currently processed LR service of the corresponding AREP.

**CurrentSlotNumber**
(Array of Unsigned8)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

This variable stores the slot number of the currently processed LR service of the corresponding AREP.

### 8.3.3 FSPMM2 state table

Table 43 contains the complete description of the FSPMM2 state machine.

**Table 43 – FSPMM2 state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | Load ARL DP master CI2.req (List of ARL Entries)<br>/valid parameters<br>=><br>Load ARL DP master CI2.cnf(+) | POWER-ON |
| 2 | POWER-ON | Load ARL DP master CI2.req (List of ARL Entries)<br>/invalid parameters<br>=><br>Status := NO<br>Load ARL DP master CI2.cnf(-) (Status) | POWER-ON |
| 3 | POWER-ON | Get ARL DP master CI2.req<br>/ARL loaded<br>=><br>Get ARL DP master CI2.cnf(+) ( List of ARL Entries ) | POWER-ON |
| 4 | POWER-ON | Get ARL DP master CI2.req<br>/ARL not loaded<br>=><br>Status := NO<br>Get ARL DP master CI2.cnf(-) (Status) | POWER-ON |
| 5 | POWER-ON | Load CRL DP master CI2.req ( List of CRL Entries )<br>/valid parameters<br>=><br>Load CRL DP master CI2.cnf(+) | POWER-ON |
| 6 | POWER-ON | Load CRL DP master CI2.req ( List of CRL Entries )<br>/invalid parameters<br>=><br>Status := NO<br>Load CRL DP master CI2.cnf(-) (Status) | POWER-ON |
| 7 | POWER-ON | Get CRL DP master CI2.req<br>/CRL loaded<br>=><br>Get CRL DP master CI2.cnf(+) (List of CRL Entries) | POWER-ON |
| 8 | POWER-ON | Get CRL DP master CI2.req<br>/CRL not loaded<br>=><br>Status := NO<br>Get CRL DP master CI2.cnf(-) (Status) | POWER-ON |
| 9 | PON | FSPMM2_Init DP master CI2 .req(Bus_Para)<br>=><br>StoreNumberIssuedMInit()<br>Current Extended Function Num[AREP]=No_Service<br>LR_State[AREP]:=W-LR-REQ<br>DMPMM2_Minit_DLL.req(Bus_Para)<br>MMAC_MInit_MM.req<br>MSAC2M_MInit_MS2.req(First_MS2_AREP ..Last_MS2_AREP) | INIT-DLM |
| 10 | INIT-DLM | DMPMM2_Minit_DLL.cnf<br>=><br>ignore | INIT-DLM |
| 11 | INIT-DLM | MMAC2_MInit_MM.cnf<br>=><br>ignore | INIT-DLM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 12 | INIT-DLM | MSAC2M_MInit_MS2.cnf (CREP)<br>/IsLastWaitingMInit()=FALSE<br>=><br>ignore | INIT-DLM |
| 13 | INIT-DLM | MSAC2M_MInit_MS2.cnf (CREP)<br>/IsLastWaitingMInit()=TRUE<br>=><br>FSPMM2_Init DP master CI2 .cnf | RUN |
| 14 | RUN | Any FSPMM2.req<br>/wrong AREP.AR_Type<br>=><br>ignore | RUN |
| 15 | RUN | Any MMAC1.cnf<br>/SetContext(Rem_add, Service) = FALSE<br>=><br>ignore | RUN |
| 16 | RUN | Any MSAC2M.req<br>/SetContext(CREP, Service) = FALSE<br>=><br>ignore | RUN |
| 17 | RUN | Any DMPMM2.req<br>/SetContext(Rem_add, Service) = FALSE<br>=><br>ignore | RUN |
| 18 | RUN | FSPMM2_Abort.req(AREP, Subnet, Instance, Reason_Code)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Abort.req | RUN |
| 19 | RUN | FSPMM2_Abort.req(AREP, Subnet, Instance, Reason_Code)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Abort.req | RUN |
| 20 | RUN | FSPMM2_Abort.req(AREP, Subnet, Instance, Reason_Code)<br>/AREP.AR_Type = MS2<br>=><br>Current Extended Function Num[AREP]=No_Service<br>LR_State[AREP]:=W-LR-REQ<br>MSAC2M_Abort.req(CREP, Subnet, Instance, Reason_Code) | RUN |
| 21 | RUN | MSAC2M_Abort.ind(CREP, Locally_Generated, Subnet, Instance, Reason_Code, Abort_Detail)<br>/SetContext(CREP, Service) = TRUE<br>=><br>Current Extended Function Num[AREP]=No_Service<br>LR_State[AREP]:=W-LR-REQ<br>FSPMM2_Abort.ind(AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) | RUN |
| 22 | RUN | DMPMM2_Event.ind(Event, Add_info)<br>=><br>FSPMM2_Event.ind(Event=Event/Fault, Add_Info=Add_info) | RUN |
| 23 | RUN | FSPMM2_Read_Slave_Diag.req(AREP)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Read_Slave_Diag.req(CREP.Rem_Add) | RUN |
| 24 | RUN | DMPMM2_Read_Slave_Diag.cnf(+)(Rem_Add, Diag_Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Slave_Diag.cnf(+)(AREP, Diag_Data) | RUN |
| 25 | RUN | DMPMM2_Read_Slave_Diag.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Slave_Diag.cnf(-)(AREP, Status=Status) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 26 | RUN | FSPMM2_Get_Cfg.req(AREP)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Get_Cfg.req(CREP.Rem_Add) | RUN |
| 27 | RUN | DMPMM2_Get_Cfg.cnf(+)(Rem_Add, Cfg_Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Get_Cfg.cnf(+)(AREP, Cfg_Data) | RUN |
| 28 | RUN | DMPMM2_Get_Cfg.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Get_Cfg.cnf(-)(AREP, Status) | RUN |
| 29 | RUN | FSPMM2_Read_Input.req(AREP)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Read_Input.req(CREP.Rem_Add) | RUN |
| 30 | RUN | DMPMM2_Read_Input.cnf(+)(Rem_Add, Inp_Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Input.cnf(+)(AREP, Inp_Data) | RUN |
| 31 | RUN | DMPMM2_Read_Input.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Input.cnf(-)(AREP, Status) | RUN |
| 32 | RUN | FSPMM2_Read_Output.req(AREP)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Read_Output.req(CREP.Rem_Add) | RUN |
| 33 | RUN | DMPMM2_Read_Output.cnf(+)(Rem_Add, Outp_Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Output.cnf(+)(AREP, Outp_Data) | RUN |
| 34 | RUN | DMPMM2_Read_Output.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Output.cnf(-)(AREP, Status) | RUN |
| 35 | RUN | FSPMM2_Set_Slave_Add.req(AREP, New_Slave_Add, Ident_Number, No_Add_Chg, Rem_Slave_Data)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Set_Slave_Add.req(CREP.Rem_Add, New_Slave_Add, Ident_Number, No_Add_Chg, Rem_Slave_Data) | RUN |
| 36 | RUN | DMPMM2_Set_Slave_Add.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Set_Slave_Add.cnf(+)(AREP) | RUN |
| 37 | RUN | DMPMM2_Set_Slave_Add.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Set_Slave_Add.cnf(-)(AREP, Status=Status) | RUN |
| 38 | RUN | Any DMPMM2.req<br>/SetContext(Rem_add, Service) = FALSE<br>=><br>ignore | RUN |
| 39 | RUN | FSPMM2_Initiate.req(AREP, Send Timeout, Features Supported, Profile Features Supported,<br>Profile Ident Number, Add Addr Param)<br>/AREP.AR_Type = MS2<br>=><br>MSAC2M_Initiate.req(CREP, Send Timeout, Features Supported, Profile Features Supported,<br>Profile Ident Number, Add Addr Param) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 40 | RUN | MSAC2M_Initiate.cnf(+)(CREP, Max Len Data Unit, Features Supported, Profile Features Supported,<br>Profile Ident Number, Add Addr Param)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Initiate.cnf(+)(AREP, Max Len Data Unit, Features Supported, Profile Features Supported,<br>Profile Ident Number, Add Addr Param) | RUN |
| 41 | RUN | MSAC2M_Initiate.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Initiate.cnf(-)(AREP, Error Decode, Error Code 1, Error Code 2) | RUN |
| 42 | RUN | FSPMM2_Read.req(AREP, Slot Number, Index, Length)<br>/AREP.AR_Type = MS2<br>=><br>MSAC2M_Read.req(CREP, Slot Number, Index, Length) | RUN |
| 43 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Read.cnf(+)(AREP, Length, Data) | RUN |
| 44 | RUN | MSAC2M_Read.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Read.cnf(-)(AREP, Error Decode, Error Code 1, Error Code 2) | RUN |
| 45 | RUN | FSPMM2_Write.req(AREP, Slot Number, Index, Length, Data)<br>/AREP.AR_Type = MS2<br>=><br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data) | RUN |
| 46 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Write.cnf(+)(AREP, Length) | RUN |
| 47 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Write.cnf(-)(AREP, Error Decode, Error Code 1, Error Code 2) | RUN |
| 48 | RUN | FSPMM2_Data_Transport.req(AREP, Slot Number, Index, Length, Data)<br>/AREP.AR_Type = MS2<br>=><br>MSAC2M_Data_Transport.req(CREP, Slot Number, Index, Length, Data) | RUN |
| 49 | RUN | MSAC2M_Data_Transport.cnf(+)(CREP, Length, Data)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Data_Transport.cnf(+)(AREP, Length, Data) | RUN |
| 50 | RUN | MSAC2M_Data_Transport.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Data_Transport.cnf(-)(AREP, Error Decode, Error Code 1, Error Code 2) | RUN |
| 51 | RUN | MSAC2M_Closed.ind(CREP)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2 DP master CI2 _Closed.ind(AREP) | RUN |
| 52 | RUN | MMAC2_Reject.ind(Rem_Add, Reason_Code)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2 DP master CI2 _Reject.ind(AREP, Reason) | RUN |
| 53 | RUN | FSPMM2_Get_Master_Diag.req(AREP, MDiag Identifier)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Get_Master_Diag.req(AREP, MDiag Identifier) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 54 | RUN | MMAC2_Get_Master_Diag.cnf(+)(Rem_Add, Diagnosis Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Get_Master_Diag.cnf(+)(AREP, Diagnosis Data) | RUN |
| 55 | RUN | MMAC2_Get_Master_Diag.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Get_Master_Diag.cnf(-)(AREP, Status) | RUN |
| 56 | RUN | FSPMM2_Start_Seq.req(AREP, Area Code, Timeout)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Start_Seq.req(AREP, Area Code, Timeout) | RUN |
| 57 | RUN | MMAC2_Start_Seq.cnf(+)(Rem_Add, Max Len Data Unit)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Start_Seq.cnf(+)(AREP, Max Len Data Unit) | RUN |
| 58 | RUN | MMAC2_Start_Seq.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Start_Seq.cnf(-)(AREP, Status) | RUN |
| 59 | RUN | FSPMM2_End_Seq.req(AREP)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_End_Seq.req(AREP) | RUN |
| 60 | RUN | MMAC2_End_Seq.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_End_Seq.cnf(+)(AREP) | RUN |
| 61 | RUN | MMAC2_End_Seq.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_End_Seq.cnf(-)(AREP, Status) | RUN |
| 62 | RUN | FSPMM2_Download.req(AREP, Area Code, Add Offset, Data)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Download.req(AREP, Area Code, Add Offset, Data) | RUN |
| 63 | RUN | MMAC2_Download.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Download.cnf(+)(AREP) | RUN |
| 64 | RUN | MMAC2_Download.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Download.cnf(-)(AREP, Status) | RUN |
| 65 | RUN | FSPMM2_Upload.req(AREP, Area Code, Add Offset, Data Len)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Upload.req(AREP, Area Code, Add Offset, Data Len) | RUN |
| 66 | RUN | MMAC2_Upload.cnf(+)(Rem_Add, Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Upload.cnf(+)(AREP, Data) | RUN |
| 67 | RUN | MMAC2_Upload.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Upload.cnf(-)(AREP, Status) | RUN |
| 68 | RUN | FSPMM2_Act_Param.req(AREP, Area Code, Activate)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Act_Param.req(AREP, Area Code, Activate) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 69 | RUN | MMAC2_Act_Param.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Act_Param.cnf(+)(AREP) | RUN |
| 70 | RUN | MMAC2_Act_Param.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Act_Param.cnf(-)(AREP, Status) | RUN |
| 71 | RUN | FSPMM2_Act_Param_Brct.req(AREP, Area Code)<br>/AREP.AR_Type = MM2<br>=><br>MMAC2_Act_Param_Brct.req(AREP, Area Code, Activate) | RUN |
| 72 | RUN | MMAC2_Act_Param_Brct.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Act_Param_Brct.cnf(+)(AREP) | RUN |
| 73 | RUN | MMAC2_Act_Param_Brct.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Act_Param_Brct.cnf(-)(AREP, Status) | RUN |
| 74 | any state | DMPMM2_SYCL_Time_Event.cnf (Send Delay Time, Status)<br>=><br>FSPMM2_SYCL_Time_Event.cnf (AREP, Send Delay Time, Status) | SAME |
| 75 | any state | FSPMM2_SYCL_Time_Event.req (AREP)<br>=><br>DMPMM2_SYCL_Time_Event.req | SAME |
| 76 | any state | FSPMM2_SYCL_Clock_Value.req (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status)<br>=><br>DMPMM2_SYCL_Clock_Value.req (Clock Value Time Event, Clock Value previous TE, Clock Value Status) | SAME |
| 77 | any state | DMPMM2_SYCL_Clock_Value.cnf (Status)<br>=><br>FSPMM2_SYCL_Clock_Value.cnf (AREP, Status) | SAME |
| 78 | any state | DMPMM2_SYCL_Clock_Value.ind (Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg)<br>=><br>FSPMM2_SYCL_Clock_Value.ind (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg) | SAME |
| 79 | any state | FSPMM2_Reset DP master Cl2 .req<br>=><br>StoreNumberIssuedResets()<br>FSPMM2_UserReset = TRUE<br>DMPMM2_Reset.req<br>MMAC2_Reset.req<br>MSAC2M_Reset.req(First_MS2_AREP ..Last_MS2_AREP) | WAIT-RESET |
| 80 | any state | DMPMM2_Fault.ind<br>=><br>StoreNumberIssuedResets()<br>FSPMM2_UserReset = FALSE<br>DMPMM2_Reset.req<br>MMAC2_Reset.req<br>MSAC2M_Reset.req(First_MS2_AREP ..Last_MS2_AREP)<br>FSPMM2 DP master Cl2 Fault.ind | WAIT-RESET |
| 81 | any state | MMAC2_Fault.ind<br>=><br>StoreNumberIssuedResets()<br>FSPMM2_UserReset = FALSE<br>DMPMM2_Reset.req<br>MMAC2_Reset.req<br>MSAC2M_Reset.req(First_MS2_AREP ..Last_MS2_AREP)<br>FSPMM2 DP master Cl2 Fault.ind | WAIT-RESET |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 82 | WAIT-RESET | DMPMM2_Reset.cnf<br>=><br>no action | WAIT-RESET |
| 83 | WAIT-RESET | MMAC2_Reset.cnf<br>=><br>no action | WAIT-RESET |
| 84 | WAIT-RESET | MSAC2M_Reset.cnf (CREP)<br>/IsLastWaitingReset()=FALSE<br>=><br>no action | WAIT-RESET |
| 85 | WAIT-RESET | MSAC2M_Reset.cnf (CREP)<br>/IsLastWaitingReset()=TRUE<br>AND FSPMM2_UserReset = TRUE<br>=><br>FSPMM2_Reset DP master CI2 .cnf | PON |
| 86 | WAIT-RESET | MSAC2M_Reset.cnf (CREP)<br>/IsLastWaitingReset()=TRUE<br>AND FSPMM2_UserReset = FALSE<br>=><br>no action | PON |
| 87 | RUN | FSPMM2_Initiate_Load.req ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Initiate_Load<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=10+sizeof(User Specific)<br>Data.w.IL.Extended_Function_Num:=Initiate_Load<br>Data.w.IL.LR_Index:=LR Index<br>Data.w.IL.Load_Type:=Load Type<br>Data.w.IL.Load_Image_Size:=Load Image Size<br>Data.w.IL.User_Specific:=User Specific<br>Data.w.IL.Intersegment_Request_Timeout:=Intersegment Request Timeout<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 88 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Initiate_Load<br>=><br>Slot_Number:=Current Slot Number[AREP]<br>Index:=255<br>Length:=10<br>MSAC2M_Read.req(CREP, Slot Number, Index, Length)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 89 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Initiate_Load<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Actual LR Size:=Data.r.IL.Actual_LR_Size<br>Max Response Delay:=Data.r.IL.Max_Response_Delay<br>Max Segment Length:=Data.r.IL.Max_Segment_Length<br>User Specific:=Data.r.IL.User_Specific<br>FSPMM2_Initiate_Load.cnf(+) ( AREP, Actual LR Size, Max Response Delay, Max Segment Length, User Specific )<br>LR_State[AREP]:=W-LR-REQ | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 90 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Initiate_Load<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Initiate_Load.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 91 | RUN | FSPMM2_Pull_Segment.req ( AREP, Slot Number, LR Index, Segment Length )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Pull<br>Current LR Index[AREP]:=LR Index<br>Index:=255<br>Length:=Segment Length<br>MSAC2M_Read.req(CREP, Slot Number, Index, Length)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 92 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Pull<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Segment Number:=Data.PULL.Sequence Number<br>More Follows:=Data.PULL.Options.More Follows<br>Data:=Data.PULL.Region Data<br>Segment Length:=Length-6<br>FSPMM2_Pull_Segment.cnf(+) ( AREP, Segment Length, Segment Number, More Follows, Data )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 93 | RUN | MSAC2M_Read.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(Service, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Pull<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Pull_Segment.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 94 | RUN | FSPMM2_Push_Segment.req ( AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Push<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=6+Segment Length<br>Data.PUSH.Extended_Function_Num:=Push<br>Data.PUSH.LR_Index:=LR Index<br>Data.PUSH.Segment_Number:=Segment Number<br>Data.PUSH.Option.More_Follows:=More Follows<br>Data.PUSH.Data:=Data<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 95 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Push<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Segment Number:=Data.PUSH.Sequence Number<br>More Follows:=Data.PUSH.Options.More Follows<br>Data:=Data.PUSH.Region Data<br>Segment Length:=Length-6<br>FSPMM2_Push_Segment.cnf(+) ( AREP)<br>LR_State[AREP]:=W-LR-REQ | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 96 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Push<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Push_Segment.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 97 | RUN | FSPMM2_Terminate_Load.req ( AREP, Slot Number, LR Index )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Terminate_Load<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=6<br>Data.TL.Extended_Function_Num:=Terminate_Load<br>Data.TL.LR_Index:=LR Index<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 98 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Terminate_Load<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Terminate_Load.cnf(+) ( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 99 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Terminate_Load<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Terminate_Load.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 100 | RUN | FSPMM2_Call.req ( AREP, Slot Number, Entity Number, FI Index, Execution Argument)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Call<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.w.CALL.Extended_Function_Num:=Call<br>Data.w.CALL. Entity_Number:= Entity_Number<br>Data.w.CALL.FI_Index:=FI Index<br>Data.w.CALL.Execution_Argument:=Execution Argument<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 101 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Call<br>=><br>Slot_Number:=Current Slot Number[AREP]<br>Index:=255<br>Length:=AREP.Max_PDU_Length<br>MSAC2M_Read.req(CREP, Slot Number, Index, Length)<br>LR_State[AREP]:=W-LR-CNF | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 102 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Call<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Result Argument:=Data.r.CALL.Result_Argument<br>FSPMM2_Call.cnf(+) ( AREP, Result Argument )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 103 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Call<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Call.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 104 | RUN | FSPMM2_Start.req ( AREP, Slot Number, FI Index, Execution Argument)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Start<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.FIS.Extended_Function_Num:=Start<br>Data.FIS.FI_Index:=FI Index<br>Data.FIS.Execution_Argument:=Execution Argument<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 105 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Start<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Start.cnf(+)( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 106 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Start<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Start.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 107 | RUN | FSPMM2_Stop.req ( AREP, Slot Number, FI Index)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Stop<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.FIS.Extended_Function_Num:=Stop<br>Data.FIS.FI_Index:=FI Index<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 108 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Stop<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Stop.cnf(+) ( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 109 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Stop<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Stop.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 110 | RUN | FSPMM2_Resume.req ( AREP, Slot Number, FI Index, Execution Argument)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Resume<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.FIS.Extended_Function_Num:=Resume<br>Data.FIS.FI_Index:=FI Index<br>Data.FIS.Execution_Argument:=Execution Argument<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 111 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Resume<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Resume.cnf(+) ( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 112 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Resume<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Resume.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 113 | RUN | FSPMM2_Reset.req ( AREP, Slot Number, FI Index)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Reset<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.FIS.Extended_Function_Num:=Reset<br>Data.FIS.FI_Index:=FI Index<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 114 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Reset<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Reset.cnf(+) ( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 115 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Reset<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Reset.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 116 | RUN | FSPMM2_Get_FI_State.req ( AREP, Slot Number, FI Index )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Get_State<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.w.STATE.Extended_Function_Num:=Get_State<br>Data.w.STATE.FI_Index:=FI Index<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 117 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Get_State<br>=><br>Slot_Number:=Current Slot Number[AREP]<br>Index:=255<br>Length:=5<br>MSAC2M_Read.req(CREP, Slot Number, Index, Length)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 118 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Get_State<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FI State:=Data.r.STATE.State<br>FSPMM2_Get_FI_State.cnf (+) ( AREP, FI State )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 119 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Get_State<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Get_State.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |

### 8.3.4 Functions

Table 44 contains the functions used by the FSPMM2, their arguments and their descriptions.

**Table 44 – Functions used by the FSPMM2**

| Function name | Description |
|---|---|
| SetContext(Rem_Add/CREP, Service) | This function checks if there exist an entry for the inputs Rem_Add/CREP and Service in the ARL and related CRL of the DP-master (Class 2).<br>A) If there exists an entry<br>then it returns TRUE and sets the local context according to the current CREP and AREP.<br>B) Otherwise it returns FALSE. |
| IsLastWaitingReset() | This function returns TRUE if all Reset service confirmation are received otherwise it returns FALSE. |
| StoreNumberIssuedResets() | This function checks stores the number of issued Reset service request in a local variable. |
| IsLastWaitingMInit() | This function returns TRUE if all MInit service confirmation are received otherwise it returns FALSE. |
| StoreNumberIssuedMInit() | This function checks stores the number of issued MInit service request in a local variable. |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# 9 Application relationship protocol machines (ARPMs)

## 9.1 MSCY1S

### 9.1.1 Primitive definitions

#### 9.1.1.1 Primitives exchanged between MSCY1S and FSPMS

Table 45 shows the service primitives including their associated parameters issued by FSPMS and received by the MSCY1S.

**Table 45 – Primitives issued by FSPMS to MSCY1S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MS0.req | FSPMS | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMS | (none) | |
| Abort.req | FSPMS | (none) | |
| Application Ready.req | FSPMS | (none) | |
| Check User Prm Result.req | FSPMS | Prm_OK | |
| Check Ext User Prm Result.req | FSPMS | Ext_Prm_OK | |
| Check Cfg Result.req | FSPMS | Cfg_OK,<br>Input Data Len,<br>Output Data Len | |
| Set Cfg.req | FSPMS | Cfg Data | |
| Set Slave Diag.req | FSPMS | Ext Diag Flag,<br>Ext Diag Overflow,<br>Ext Diag Data | |
| Set Input.req | FSPMS | Input Data | |
| Get Output.req | FSPMS | (none) | |

Table 46 shows the service primitives including their associated parameters issued by MSCY1S and received by the FSPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 46 – Primitives issued by MSCY1S to FSPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MS0.cnf | MSCY1S | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | MSCY1S | (none) | |
| Check User Prm Result.cnf(+) | MSCY1S | (none) | |
| Check User Prm Result.cnf(-) | MSCY1S | Status | |
| Check Ext User Prm Result.cnf(+) | MSCY1S | (none) | |
| Check Ext User Prm Result.cnf(-) | MSCY1S | Status | |
| Check Cfg Result.cnf(+) | MSCY1S | (none) | |
| Check Cfg Result.cnf(-) | MSCY1S | Status | |
| Set Cfg.cnf(+) | MSCY1S | (none) | |
| Set Cfg.cnf(-) | MSCY1S | (none) | |
| Set Slave Diag.cnf(+) | MSCY1S | (none) | |
| Set Slave Diag.cnf(-) | MSCY1S | (none) | |
| Set Input.cnf(+) | MSCY1S | (none) | |
| Set Input.cnf(-) | MSCY1S | (none) | |
| Get Output.cnf(+) | MSCY1S | Output Data,<br>Clear Flag,<br>New Flag | |
| Get Output.cnf(-) | MSCY1S | (none) | |
| Start.ind | MSCY1S | Actual Enabled Alarms,<br>Alarm Sequence,<br>Alarm Limit | |
| Stop.ind | MSCY1S | (none) | |
| Fault.ind | MSCY1S | (none) | |
| Set Slave Add.ind | MSCY1S | New Slave Add,<br>Ident Number,<br>No Add Chg,<br>Rem Slave Data | |
| Check User Prm.ind | MSCY1S | User Prm Data | |
| Check Ext User Prm.ind | MSCY1S | Ext User Prm Data | |
| Check Cfg.ind | MSCY1S | Check Cfg Mode,<br>Cfg Data | |
| New Output.ind | MSCY1S | Clear Flag | |
| SYNCH Event.ind | MSCY1S | Status | |
| Global Control.ind | MSCY1S | Clear Command,<br>Sync Command,<br>Freeze Command,<br>Group Select | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 9.1.1.2 Parameters of MSCY1S primitives

The parameters used with the primitives exchanged between the FSPMS and the MSCY1S are described in FAL Service Definition in IEC 61158-5-3.

### 9.1.2 State machine description

The MSCY1S State Machine handles the following services issued by the DP-master (Class 1):

diagnostic,

parameterisation,

configuration,

(cyclic) data exchange.

The MSCY1S State Machine starts (Start.req/cnf) and stops (Stop.req/cnf) the MSAC1S State Machine that handles acyclic services. The MSAC1S State Machine sends an Abort.ind to the MSCY1S State Machine that closes also the cyclic connection. The MSCY1S State Machine informs the FSPMS that the MSAC1S and the MSCY1S have been started (Start.ind) or stopped (Stop.ind).

If a Slave does not implement the parameterization structures DPV1_Status 1-3, the MSAL1S and MSAC1S State Machines, the following macros should be left empty. In this case the variable Operation_Mode remains in state V0:

PV0V0 shall return TRUE

OPERATION_MODE_OK shall return TRUE

NOPRMCMD shall return TRUE

ISO_MODE_OK shall return TRUE

STOP_ISOM

STOP_ASM

CHECK_ALARM_START

SET_OPERATION_MODE

SET_ALARM_CHKCFGM

EXE_PRM_CMD

START_C1

STOP_C1

START_C1_CON

STOP_C1_CON

**Rules for checking parameterization data**

The rules according to Table 47 shall be used to check the octets DPV1_Status_1 to DPV1_Status_3 from the Set_Prm-REQ-PDU with local capabilities of the DP-slave.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 47 – Rules for DPV1_Status_1, DPV1_Status_2 and DPV1_Status_3 check**

| Bit(s) in Set_Prm-REQ-PDU | Rules for checking | |
|---|---|---|
| Reserved bits in field DPV1_Status_1 ( Bit 0, Bit 1, Bit 3, Bit 4) | These bits shall not be checked (don't care).<br>=> okay for any combination | |
| Bit WD_Base_1ms in field DPV1_Status_1 | = FALSE(0):<br>=> okay | = TRUE (1) & WD_Base_1ms_supp = 1:<br>=> okay<br>= TRUE (1) & WD_Base_1ms_supp = 0:<br>=> okay, but the DP-slave shall load it's local WD-Timer with a corrected value (based on the next possible value with 10 ms base) |
| Bit Publisher_Enable in field DPV1_Status_1 | = FALSE(0):<br>=> okay | = TRUE(1) & Publisher_supp = 1:<br>=> okay<br>= TRUE(1) & Publisher_supp = 0:<br>=> Prm_Fault |
| Bit Fail_Safe in field DPV1_Status_1 | = FALSE(0) & Fail_Safe_required = 1:<br>=> Prm_Fault<br>= FALSE(0) & Fail_Safe_required = 0:<br>=> okay | = TRUE(1) & Fail_Safe_supp = 1:<br>=> okay<br>= TRUE(1) & Fail_Safe_supp = 0:<br>=> Prm_Fault |
| Bit DPV1_Enable in field DPV1_Status_1 | = FALSE(0) & C1_Read_Write_required=0:<br>=> okay<br>= FALSE(0) & C1_Read_Write_required=1:<br>=> Prm_Fault | = TRUE(1) & C1_Read_Write_supp = 0:<br>=> Prm_Fault<br>= TRUE(1) & C1_Read_Write_supp = 1:<br>=> okay, open MS1 application relationship |
| Bit Check_Cfg_Mode in field DPV1_Status_2 | = 0 (strict check) :<br>=> okay<br>The application process has to check the data of the following Chk_Cfg-REQ-PDU in a strict way (equality). | = 1 (more flexible check):<br>=> okay;<br>The application process has to check the data of the following Chk_Cfg-REQ-PDU in a more flexible way (compatibility). |
| Reserved Bit 1 in field DPV1_Status_2 | = 0:<br>=> okay | = 1:<br>=> Prm_Fault |
| Enable_xx_Alarm Bits in field DPV1_Status_2 | = FALSE(0) & xx_Alarm_required = 1 & DPV1_Status_1.DPV1_Enable = FALSE (0):<br>=> Prm_Fault<br>= FALSE(0) & xx_Alarm_required = 0 & DPV1_Status_1.DPV1_Enable = FALSE (0):<br>=> okay<br>= FALSE(0) & xx_Alarm_required = 1 & DPV1_Status_1.DPV1_Enable = TRUE (1):<br>=> Prm_Fault<br>= FALSE(0) & xx_Alarm_required = 0 & DPV1_Status_1.DPV1_Enable = TRUE (1):<br>=> okay | =TRUE(1) & DPV1_Status_1.DPV1_Enable = FALSE(0):<br>=> Prm_Fault<br>=TRUE(1) & DPV1_Status_1.DPV1_Enable = TRUE (1):<br>=> okay |
| Bits Alarm_Mode in field DPV1_Status_3 | 0<br><br>=> okay | <>0 & Alarm_Sequence_Mode_count<>0:<br>=> okay<br><>0 & Alarm_Sequence_Mode_count=0:<br>=> Prm_Fault |
| Bit Prm_Structure in field DPV1_Status_3 | = FALSE(0) & Prm_Structure_required=FALSE:<br>=>okay<br>= FALSE(0) & Prm_Structure_required=TRUE:<br>=> Prm_Fault | = TRUE(1) & Prm_Structure_supp=TRUE:<br>=>okay<br><br>= TRUE(1) & Prm_Structure_supp=FALSE:<br>=> Prm_Fault |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Bit(s) in Set_Prm-REQ-PDU | Rules for checking | |
|---|---|---|
| Bit IsoM_Req in field DPV1_Status_3 | = FALSE(0) & Isochronous_Mode_required=FALSE: =>okay<br>= FALSE(0) & Isochronous_Mode_required=TRUE: => Prm_Fault | = TRUE(1) & (Isochronous_Mode_supp = TRUE & DPV1_Status_1.Fail_Safe = TRUE & Group_Ident< 0x80 & Freeze_Req=0 & Sync_Req=0): => okay<br>TRUE(1) & !(Isochronous_Mode_supp = TRUE & DPV1_Status_1.Fail_Safe = TRUE & Group_Ident< 0x80 & Freeze_Req=0 & Sync_Req=0): => Prm-Fault |
| Bit PrmCmd in field DPV1_Status_3 | = FALSE(0) & PrmCmd_required=FALSE: =>okay<br>= FALSE(0) & PrmCmd_required=TRUE:<br>=> Prm_Fault | = TRUE(1) & (PrmCmd_supp = TRUE): => okay<br>TRUE(1) & (PrmCmd_supp = FALSE): => Prm-Fault |
| Reserved-Bits (Bit 5 and Bit 6) in field DPV1_Status_3 | = 0: => okay | 1: => Prm_Fault |
| NOTE The table above gives a comprehensive overview on all checks that belong to the parameter. However, some of them may also checked on other places within the state machine. | | |

**Isochronous behaviour**

Devices that support isochronous operation shall set the concerning AR attributes according to the configuration. The block for isochronous parameters belong to the application and are part of user data within the Set_Prm-REQ-PDU or Set_Ext_Prm-REQ-PDU. The DP-master sets the DP-slave into isochronous operation by setting the IsoM_Req-Bit TRUE. When the MSCY1S receives the Set_Prm-REQ-PDU it has to check the following condition:

If(Isochronous Mode = TRUE) then Failsafe shall be set and any Group shall not be set concerning the Set_Prm-REQ-PDU otherwise the DP-slaves shall generate a Prm_Fault and shall leave the Data Exchange state. DP-slaves that do not support isochronous operation may be integrated in an isochronous working system by selecting “Group_8” for only this DP-slaves and using Sync and Freeze commands.

If the condition above is fulfilled the DP-slave operates isochronous. Additionally to normal cyclic data exchange the MSCY1S generates a SYNCH Event.ind with Status “IsoM Start” with receiving the first Global_Control-REQ-PDU, “IsoM SYNCH” with subsequent Global_Control-REQ-PDU’s, and “IsoM Stop” when receiving Global_Control-REQ-PDU with Control Command “Clear” or leaving the data exchange cycle. This service triggers the PLL of the application process.The right access with Get Input and Set Output from the application point of view is monitored with the help of the PLL.

**Redundancy behaviour**

Devices that support Slave redundancy shall support the PrmCmd. This allows a DP-master (Class 1) to use a DP-slave as Primary or Backup. Aditionally the DP-master (Class 1) can stop and start MS1 after a swichover of the DP-master. MSCY1S provides the basic mechanism to support a redundancy structure in the Application.

**Local variables**

**Act_Ref**
Counter for Reference Numbers for Diagnosis Updates.

**Act_Cnt**
Storage for Users Reference number

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Ref_Cnt**
Storage of Act_Ref at users Diagnosis.

**B Sync**
(Octet-String)

Intermediate storage for outputs if the DP-slave is operated in Sync Mode.

**B Output**
(Octet-String)

Output data of the DP-slave that can be fetched by the Get Output service. The requirements for the consistency as described in the Cfg_Data shall be taken into consideration.

**B Freeze**
(Octet-String)

Intermediate storage for input data.

**B Input**
(Octet-String)

Depending on the type (see description of Cfg_Data), the data shall be consistently loaded by the Set Input service, or may be captured freely.

**B Real_Cfg**
(Octet-String)

The configuration which was previously defined for the device, or the configuration which is determined in the start-up by the device, respectively. The structure of the individual octets corresponds to the structure of Cfg_Data (->Chk_Cfg).

**B User Prm**
(Octet-String)

Intermediate storage for User Prm data.

**B Ext User Prm**
(Octet-String)

Intermediate storage for Ext User Prm data.

**B Cfg**
(Octet-String)

Intermediate storage for Cfg data.

**Diag**
(Octet-String)

Intermediate storage for Diag data.

**Ext Diag Data**
(Octet-String)

Intermediate storage for Ext Diag Data, the diagnosis setup by the user.

**Ext Diag Flag**
(Boolean)

Intermediate storage for Ext Diag Flag, which indicates important user diagnosis.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Clear Command**
**Sync Command**
**Freeze Command**
Intermediate storage for global control commands.

**Station_Address**
(Unsigned8)

Own station address which may be set by the service "Set_Slave_Add" or which is locally stored.

**Check_Prm_Add**
(Unsigned8)

Masters station address which has invoked the last service "Set_Prm ".

**Check_Ext_Prm_Add**
(Unsigned8)

Masters station address which has invoked the last service "Set_Ext_Prm ".

**Check_Cfg_Add**
(Unsigned8)

Masters station address which has invoked the last service "Chk_Cfg ".

**Output Data Len**
(Unsigned8)

Indicates the number of output octets that are actually used.

**Input Data Len**
(Unsigned8)

Indicates the number of input octets that are actually used.

**Real_No_Add_Chg**
(Boolean)

Indicates if an address change is possible (FALSE) or not. Can be statically defined or stored.

**Active_Groups**
(Unsigned8)

Contains the Group_Ident as set via Set_Prm. This is used at Global_Control for the decision if the station is addressed.

**Diag_Flag**
(Boolean)

This flag indicates that the Master shall be informed by a high prior Reply-Update that a change in the diagnostic information occurred.

**Sync_Supp**
(Boolean)

Indicates that the DP-slave does support the Sync-Mode.

**Freeze_Supp**
(Boolean)

Indicates that the DP-slave does support the Freeze-Mode.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Operation_Mode**
(Boolean)

Local flag that stores the Operation_Mode set with the Set_Prm service. Operation_Mode=V0 means that the Slave does not support acyclic services on MSAC1 and alarms.

**Prm_Pending**
(Unsigned8)

Local flag that indicates that the User is still processing the last call of the function “Check UserPrm ".

**Ext_Prm_Pending**
(Unsigned8)

Local flag that indicates that the User is still processing the last call of the function “Check Ext_Prm ".

**Cfg_Pending**
(Unsigned8)

Local flag that indicates that the User is still processing the last call of the function “Check Cfg ".

**Input_Pending**
(Unsigned8)

Local flag that indicates that the User has not yet set the Input data

**New Output**
(Unsigned8)

Local flag that indicates that there are Output data available not yet fetched with "Get Output".

**DX_Entered**
(Boolean)

Local flag which is set after the MSCY1S State Machine has entered the DATA-EXCH mode and at least one Data_Exchange service has been completed.

**Start_State**
(Unsigned8)

Local variable which is used to store the state of the MSAC1S-State Machine.

Possible values for Start_State:

Idle MSAC1S is closed

Run MSAC1S is open

B Start_Operation is initiated

E Stop_Operation is initiated

BE Start_Operation is initiated, Stop_Operation is queued

EB Stop_Operation is initiated, Start _Operation is queued

BEB Start_Operation is initiated, Stop_Operation (first) and Start_Operation(second) are queued

**Safe_State**
(Boolean)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Indicates whether the outputs has to be switched to the safe state.

**Chk Cfg Mode**
(Boolean)

Indicates how to check the Cfg Data.

**First Synch**
(Boolean)

This local variable is FALSE if the Isochronous mode has been activated and at least one Global_Control-REQ-PDU with Group Select (Group_8) has been received.

### 9.1.3 MSCY1S state table

Table 48 contains the complete description of the MSCY1S state machine.

**Table 48 – MSCY1S state table**

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 1 | POWER-ON | MSCY1S SInit MS0.req ()<br>=><br>Diag.Ext_Diag_Data := NIL<br>Diag.Ext Diag Flag := FALSE<br>B Real Cfg := Real Cfg Data<br>Chk Cfg Mode := FALSE<br>Sync Supp := Sync Supported<br>Freeze Supp := Freeze Supported<br>Diag.Ident Number := Ident Number<br>Real No Add Chg := No Add Chg<br>Act_Ref := 0<br>Act_Cnt := 0<br>Ref_Cnt := 0<br>DMPMS Slave Init.req () | WAIT-INI-CON |
| 2 | POWER-ON | MSCY1S Abort.req ()<br>=><br>ignore | POWER-ON |
| 3 | POWER-ON | MSCY1S Set Slave Diag.req ( Ext Diag Flag, Ext Diag Overflow, Ext Diag Data, Reference )<br>=><br>Act_Cnt := Reference<br>MSCY1S Set Slave Diag.cnf(-) (Reference := Act_Cnt) | POWER-ON |
| 4 | POWER-ON | MSCY1S Set Cfg.req ( Cfg Data )<br>=><br>MSCY1S Set Cfg.cnf(-) () | POWER-ON |
| 5 | POWER-ON | MSCY1S Get Output.req ()<br>=><br>MSCY1S Get Output.cnf(-) () | POWER-ON |
| 6 | POWER-ON | MSCY1S Set Input.req ( Input Data )<br>=><br>MSCY1S Set Input.cnf(-) () | POWER-ON |
| 7 | POWER-ON | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>=><br>Status := Reset<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | POWER-ON |
| 8 | POWER-ON | MSCY1S Application Ready.req()<br>=><br>ignore | POWER-ON |
| 9 | POWER-ON | MSCY1S Check User Prm Result.req ( Prm_OK )<br>=><br>Status := Reset<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 10 | POWER-ON | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>=><br>Status := Reset<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | POWER-ON |
| 11 | WAIT-INI-CON | DMPMS Slave Init.cnf ()<br>=><br>Operation Mode := V0<br>Start_State := Idle<br>Safe_State:=TRUE<br>Diag.Prm_Req := TRUE<br>Diag.Cfg_Fault := FALSE<br>Diag.Prm_Fault := FALSE<br>Diag.Not_Supported := FALSE<br>Diag.Master_Add := Invalid<br>Diag.WD_On := FALSE<br>Diag.Station_Not_Ready := TRUE<br>Diag.Sync_Mode := FALSE<br>Diag.Freeze_Mode := FALSE<br>Diag.Stat_Diag := FALSE<br>Diag.Ext_Diag_Overflow := FALSE<br>B User Prm := NIL<br>B Cfg := NIL<br>Diag_Flag := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>Prm Structure = FALSE<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Get Cfg Upd.req ( Cfg Data )<br>MSCY1S SInit MS0.cnf () | WAIT-PRM |
| 12 | WAIT-PRM | MSCY1S Abort.req ()<br>=><br>ignore | WAIT-PRM |
| 13 | WAIT-PRM | MSCY1S Set Slave Diag.req ( Ext Diag Flag, Ext Diag Overflow, Ext Diag Data, Reference )<br>=><br>Diag.Ext Diag Data := Ext Diag Data<br>Diag.Ext Diag Flag := Ext Diag Flag<br>Diag.Ext Diag Overflow := Ext Diag Overflow<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>Act_Cnt := Reference<br>Ref_Cnt := Act_Ref<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>MSCY1S Set Slave Diag.cnf(+) (Reference := Act_Cnt ) | WAIT-PRM |
| 14 | WAIT-PRM | MSCY1S Set Cfg.req ( Cfg Data )<br>=><br>DMPMS Get Cfg Upd.req ( Cfg Data )<br>MSCY1S Set Cfg.cnf(+) () | WAIT-PRM |
| 15 | WAIT-PRM | MSCY1S Get Output.req ()<br>=><br>MSCY1S Get Output.cnf(-) () | WAIT-PRM |
| 16 | WAIT-PRM | MSCY1S Set Input.req ( Input Data )<br>=><br>MSCY1S Set Input.cnf(-) () | WAIT-PRM |
| 17 | WAIT-PRM | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | WAIT-PRM |
| 18 | WAIT-PRM | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 2<br>=><br>Status := New Prm<br>Prm_Pending := 1<br>User Parameter Data := B User Prm<br>MSCY1S Check User Prm Result.cnf(-) ( Status )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 19 | WAIT-PRM | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1<br>=><br>Prm_Pending := 0<br>MSCY1S Check User Prm Result.cnf(+)() | WAIT-PRM |
| 20 | WAIT-PRM | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | WAIT-PRM |
| 21 | WAIT-PRM | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 2<br>=><br>Status := New Cfg<br>Cfg_Pending := 1<br>Cfg Data := B Cfg<br>MSCY1S Check Cfg Result.cnf(-) ( Status )<br>MSCY1S Check Cfg.ind ( Cfg Data ) | WAIT-PRM |
| 22 | WAIT-PRM | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1<br>=><br>Cfg_Pending := 0<br>MSCY1S Check Cfg Result.cnf(+) () | WAIT-PRM |
| 23 | WAIT-PRM | MSCY1S Application Ready.req()<br>=><br>ignore | WAIT-PRM |
| 24 | WAIT-PRM | MSAC1S Abort.ind()<br>=><br>ignore | WAIT-PRM |
| 25 | WAIT-PRM | MSAC1S Start.cnf()<br>=><br>START_C1_CON | WAIT-PRM |
| 26 | WAIT-PRM | MSAC1S Stop.cnf()<br>=><br>Start_State := Idle | WAIT-PRM |
| 27 | WAIT-PRM | DMPMS Set Slave Add.ind(New Slave Add, Ident Number, No Add Chg, Rem Slave Data)<br>/Real_No_Add_Chg = FALSE && Diag.Ident_Number = Ident_Number<br>&& New_Slave_Address <= 125<br>=><br>MSCY1S Set Slave Add.ind ( New Slave Add, Ident Number, No Add Chg, Rem Slave Data ) | POWER-ON |
| 28 | WAIT-PRM | DMPMS Set Slave Add.ind(New Slave Add, Ident Number, No Add Chg, Rem Slave Data)<br>/Real_No_Add_Chg = TRUE \|\| Diag.Ident_Number <> Ident_Number<br>\|\| New_Slave_Add > 125<br>=><br>ignore | WAIT-PRM |
| 29 | WAIT-PRM | DMPMS Slave Diag.ind(Req Add, Reference)<br>=><br>ignore | WAIT-PRM |
| 30 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = FALSE && Unlock_Req = FALSE<br>=><br>DMPMS Set minTsdr.req ( MinTsdr ) | WAIT-PRM |
| 31 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE<br>=><br>ignore | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 32 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE &&<br>(Ident_Number <> Diag.Ident_Number \|\| !OPERATION_MODE_OK \|\| WD_On =<br>TRUE && (WD_Fact_1=0 \|\| WD_Fact_2=0) )<br>=><br>Diag.Prm_Fault := TRUE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 33 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE<br>&& Ident_Number = Diag.Ident_Number && OPERATION_MODE_OK &&<br>(WD_On = FALSE \|\| (WD_Fact_1 > 0 && WD_Fact_2 > 0))<br>&& (Freeze_Req = TRUE && Freeze Supp = FALSE \|\| Sync_Req = TRUE<br>&& Sync Supp = FALSE \|\| Prm_Data[1].0, .1, .2 = TRUE)<br>=><br>Diag.Not_Supported := TRUE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 34 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len < 7<br>=><br>ignore | WAIT-PRM |
| 35 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/PRM_OK && Prm_Pending = 0<br>=><br>Diag.Master_Add := Req Add<br>Check_Prm_Add := Req Add<br>FirstSynch := TRUE<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups := Group_Ident<br>Diag.Prm_Fault := FALSE<br>Diag.Prm_Req := FALSE<br>Diag.Not_Supported := FALSE<br>Prm_Pending := 1<br>User Parameter Data := Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Input_Pending := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-CFG |
| 36 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/PRM_OK && Prm_Pending > 0<br>=><br>Diag.Master_Add := Req Add<br>Check_Prm_Add := Req Add<br>FirstSynch := TRUE<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups := Group_Ident<br>Diag.Prm_Fault := FALSE<br>Diag.Prm_Req := FALSE<br>Diag.Not_Supported := FALSE<br>Prm_Pending := 2<br>B User Prm := Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Input_Pending := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | WAIT-CFG |
| 37 | WAIT-PRM | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>=><br>ignore | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 38 | WAIT-PRM | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add = Req_Add<br>=><br>ignore | WAIT-PRM |
| 39 | WAIT-PRM | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | WAIT-PRM |
| 40 | WAIT-PRM | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 2<br>=><br>Status := New Prm<br>Prm_Pending := 1<br>Ext User Parameter Data := B Ext User Prm<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status )<br>MSCY1S Check Ext User Prm.ind ( Ext User Parameter Data ) | WAIT-PRM |
| 41 | WAIT-PRM | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1<br>=><br>Ext_Prm_Pending := 0<br>MSCY1S Check Ext User Prm Result.cnf(+)() | WAIT-PRM |
| 42 | WAIT-CFG | MSCY1S Abort.req ()<br>=><br>Operation_Mode := V0<br>Diag.Master_Add := invalid<br>Diag.Prm_Req := TRUE<br>Diag.Station_Not_Ready := TRUE<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 43 | WAIT-CFG | MSCY1S Set Slave Diag.req ( Ext Diag Flag, Ext Diag Overflow, Ext Diag Data, Reference )<br>=><br>Diag.Ext Diag Data := Ext Diag Data<br>Diag.Ext Diag Flag := Ext Diag Flag<br>Diag.Ext Diag Overflow := Ext Diag Overflow<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>Act_Cnt := Reference<br>Ref_Cnt := Act_Ref<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>MSCY1S Set Slave Diag.cnf(+) (Reference := Act_Cnt ) | WAIT-CFG |
| 44 | WAIT-CFG | MSCY1S Set Cfg.req ( Cfg Data )<br>=><br>DMPMS Get Cfg Upd.req ( Cfg Data )<br>MSCY1S Set Cfg.cnf(+) () | WAIT-CFG |
| 45 | WAIT-CFG | MSCY1S Get Output.req ()<br>=><br>MSCY1S Get Output.cnf(-) () | WAIT-CFG |
| 46 | WAIT-CFG | MSCY1S Set Input.req ( Input Data )<br>/Input_Pending = FALSE<br>=><br>MSCY1S Set Input.cnf(-) () | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 47 | WAIT-CFG | MSCY1S Set Input.req ( Input Data )<br>/Input_Pending = TRUE && Operation_Mode = V0<br>=><br>Inp Data := Input Data<br>Diag.Cfg_Fault := FALSE<br>Diag_Flag := TRUE<br>Diag.Stat_Diag := TRUE<br>Diag.Station_Not_Ready := FALSE<br>DX_Entered := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>MSCY1S Set Input.cnf(+) ()<br>DMPMS Enter.req ( Master Add, Input Data Len, Output Data Len )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | DATA-EXCH |
| 48 | WAIT-CFG | MSCY1S Set Input.req ( Input Data )<br>/Input_Pending = TRUE && Operation_Mode = V1<br>=><br>B Input := Input Data<br>START_C1<br>MSCY1S Set Input.cnf(+) () | WAIT-CFG |
| 49 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | WAIT-CFG |
| 50 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 2<br>=><br>Status := New Prm<br>Prm_Pending := 1<br>User Parameter Data := B User Prm<br>MSCY1S Check User Prm Result.cnf(-) ( Status )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-CFG |
| 51 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add <> Check_Prm_Add<br>=><br>Status := Inv Master Add<br>Prm_Pending := 0<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | WAIT-CFG |
| 52 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Prm_OK = TRUE<br>=><br>Prm_Pending := 0<br>MSCY1S Check User Prm Result.cnf(+)() | WAIT-CFG |
| 53 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Prm_OK = FALSE<br>=><br>Prm_Pending := 0<br>Diag.Prm_Fault := TRUE<br>Diag.Master_Add := Invalid<br>Diag.Prm_Req := TRUE<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>MSCY1S Check User Prm Result.cnf(+)()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 54 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 55 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 2<br>=><br>Status := New Cfg<br>Cfg_Pending := 1<br>Cfg Data := B Cfg<br>MSCY1S Check Cfg Result.cnf(-) ( Status )<br>MSCY1S Check Cfg.ind ( Cfg Data ) | WAIT-CFG |
| 56 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add <> Check_Cfg_Add<br>=><br>Status := Inv Master Add<br>Cfg_Pending := 0<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | WAIT-CFG |
| 57 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Input Data Len > 0<br>=><br>Inp Data Len := Input Data Len<br>Outp Data Len := Output Data Len<br>Cfg_Pending := 0<br>Input_Pending := TRUE<br>MSCY1S Check Cfg Result.cnf(+) () | WAIT-CFG |
| 58 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Operation_Mode = V1 && Input Data Len = 0<br>=><br>Inp Data Len := 0<br>Outp Data Len := Output Data Len<br>Inp Data := NIL<br>Cfg_Pending := 0<br>START_C1<br>MSCY1S Check Cfg Result.cnf(+) () | WAIT-CFG |
| 59 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Operation_Mode = V0 && Input Data Len = 0<br>=><br>Inp Data Len := 0<br>Outp Data Len := Output Data Len<br>Inp Data := NIL<br>Cfg_Pending := 0<br>Diag.Cfg_Fault := FALSE<br>Diag_Flag := TRUE<br>Diag.Stat_Diag := TRUE<br>Diag.Station_Not_Ready := FALSE<br>DX_Entered := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Enter.req<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>MSCY1S Check Cfg Result.cnf(+) () | DATA-EXCH |
| 60 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = FALSE<br>=><br>Cfg_Pending := 0<br>Diag.Cfg_Fault := TRUE<br>Diag.Prm_Req := TRUE<br>Diag.Master_Add := Invalid<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>MSCY1S Check Cfg Result.cnf(+) () | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 61 | WAIT-CFG | MSCY1S Application Ready.req()<br>=><br>ignore | WAIT-CFG |
| 62 | WAIT-CFG | MSAC1S Abort.ind()<br>=><br>ignore | WAIT-CFG |
| 63 | WAIT-CFG | MSAC1S Start.cnf()<br>/Start_State <> B<br>=><br>START_C1_CON | WAIT-CFG |
| 64 | WAIT-CFG | MSAC1S Start.cnf()<br>/Start_State = B<br>=><br>Diag.Cfg_Fault := FALSE<br>Diag_Flag := TRUE<br>Diag.Stat_Diag := TRUE<br>Diag.Station_Not_Ready := FALSE<br>DX_Entered := FALSE<br>START_C1_CON<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Enter.req<br>DMPMS RD Inp Upd.req ( Inp Data )<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | DATA-EXCH |
| 65 | WAIT-CFG | MSAC1S Stop.cnf()<br>=><br>STOP_C1_CON | WAIT-CFG |
| 66 | WAIT-CFG | MSAC1S Abort.ind()<br>=><br>Operation_Mode := V0<br>Diag.Master_Add := invalid<br>Diag.Prm_Req := TRUE<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 67 | WAIT-CFG | DMPMS Set Slave Add.ind(New Slave Add, Ident Number, No Add Chg, Rem Slave Data)<br>=><br>ignore | WAIT-CFG |
| 68 | WAIT-CFG | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add = Req_Add<br>=><br>TRIG_WD | WAIT-CFG |
| 69 | WAIT-CFG | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | WAIT-CFG |
| 70 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE<br>&& Diag.Master_Add = Req_Add<br>=><br>Diag.Master_Add := invalid<br>Diag.Prm_Req := TRUE<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 71 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE<br>&& Diag.Master_Add <> Req_Add<br>=><br>ignore | WAIT-CFG |
| 72 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = FALSE && Lock_Req = TRUE<br>&& Diag.Master_Add = Req_Add && (Ident_Number <> Diag.Ident_Number<br>\|\| OPERATION_MODE_OK = FALSE \|\| WD_On = TRUE && (WD_Fact_1=0 \|\|<br>WD_Fact_2=0))<br>=><br>Diag.Prm_Fault := TRUE<br>Diag.Master_Add := Invalid<br>Diag.Prm_Req := TRUE<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 73 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = FALSE && Lock_Req = TRUE<br>&& Diag.Master_Add = Req_Add && Ident_Number = Diag.Ident_Number<br>&& OPERATION_MODE_OK && (WD_On = FALSE \|\| WD_Fact_1>0 &&<br>WD_Fact_2>0)<br>&& (Freeze_Req = TRUE && Freeze Supp = FALSE<br>\|\| Sync_Req = TRUE && Sync Supp = FALSE \|\|<br>Prm_Data[1].0, .1, .2 = TRUE)<br>=><br>Diag.Not_Supported := TRUE<br>Diag.Master_Add := Invalid<br>Diag.Prm_Req := TRUE<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 74 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = FALSE && Lock_Req = TRUE<br>&& Diag.Master_Add <> Req_Add && (Ident_Number <> Diag.Ident_Number<br>\|\| OPERATION_MODE_OK = FALSE \|\| WD_On = TRUE && (WD_Fact_1=0 \|\|<br>WD_Fact_2=0)<br>\|\| Freeze_Req = TRUE && Freeze Supp = FALSE<br>\|\| Sync_Req = TRUE && Sync Supp = FALSE<br>\|\| Prm_Data[1].0, .1, .2 = TRUE )<br>=><br>ignore | WAIT-CFG |
| 75 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/PRM_OK && Prm_Pending = 0<br>=><br>Diag.Master_Add := Req_Add<br>Check_Prm_Add := Req Add<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups := Group_Ident<br>Prm_Pending := 1<br>User Parameter Data := Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Input_Pending := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 76 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/PRM_OK && Prm_Pending > 0<br>=><br>Diag.Master_Add := Req_Add<br>Check_Prm_Add := Req Add<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups := Group_Ident<br>Prm_Pending := 2<br>B User Prm := Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Input_Pending := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | WAIT-CFG |
| 77 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = FALSE && Lock_Req = FALSE<br>=><br>DMPMS Set minTsdr.req ( MinTsdr ) | WAIT-CFG |
| 78 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len < 7<br>=><br>ignore | WAIT-CFG |
| 79 | WAIT-CFG | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | WAIT-CFG |
| 80 | WAIT-CFG | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add = Req_Add && Cfg_Pending = 0<br>=><br>Cfg_Pending := 1<br>Check_Cfg_Add := Req Add<br>TRIG_WD<br>MSCY1S Check Cfg.ind ( Check Cfg Mode, Cfg Data ) | WAIT-CFG |
| 81 | WAIT-CFG | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add = Req_Add && Cfg_Pending > 0<br>=><br>Cfg_Pending := 2<br>Check_Cfg_Add := Req Add<br>B Cfg := Cfg Data<br>TRIG_WD | WAIT-CFG |
| 82 | WAIT-CFG | WDTimer expired<br>=><br>Diag.Master_Add := invalid<br>Diag.Prm_Req := TRUE<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 83 | WAIT-CFG | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add = Req_Add &&Ext_Prm_Pending = 0<br>=><br>TRIG_WD<br>Ext User Parameter Data := Ext User Prm Data<br>Ext_Prm_Pending := 1<br>Check_Ext_Prm_Add := Req_Add<br>MSCY1S Check Ext User Prm.ind ( Ext User Parameter Data ) | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 84 | WAIT-CFG | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add = Req_Add &&<br>Ext_Prm_Pending > 0<br>=><br>TRIG_WD<br>B Ext User Prm := Ext User Prm Data<br>Ext_Prm_Pending := 2<br>Check_Ext_Prm_Add := Req_Add | WAIT-CFG |
| 85 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | WAIT-CFG |
| 86 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 2<br>=><br>Status := New Prm<br>Ext_Prm_Pending := 1<br>Ext User Parameter Data := B Ext User Prm<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status )<br>MSCY1S Check Ext User Prm.ind ( Ext User Parameter Data ) | WAIT-CFG |
| 87 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add <> Check_Ext_Prm_Add<br>=><br>Status := Inv Master Add<br>Ext_Prm_Pending := 0<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | WAIT-CFG |
| 88 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add = Check_Ext_Prm_Add &&<br>Ext_Prm_OK = TRUE<br>=><br>Ext_Prm_Pending := 0<br>MSCY1S Check Ext User Prm Result.cnf(+)() | WAIT-CFG |
| 89 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Ext_Prm_OK = FALSE<br>=><br>Ext_Prm_Pending := 0<br>Diag.Cfg_Fault := TRUE<br>Diag.Master_Add := Invalid<br>Diag.Prm_Req := TRUE<br>StopTimer(WD)<br>Diag.WD_On := FALSE<br>STOP_C1<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>MSCY1S Check Ext User Prm Result.cnf(+)()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 90 | DATA-EXCH | MSCY1S Abort.req ()<br>=><br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 91 | DATA-EXCH | MSCY1S Set Slave Diag.req ( Ext Diag Flag, Ext Diag Overflow, Ext Diag Data, Reference )<br>=><br>Diag.Ext Diag Data := Ext Diag Data<br>Diag.Ext Diag Flag := Ext Diag Flag<br>Diag.Ext Diag Overflow := Ext Diag Overflow<br>Diag Data := Diag<br>Diag Flag := TRUE<br>Act_Ref := Act_Ref + 1<br>Act_Cnt := Reference<br>Ref_Cnt := Act_Ref<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Data Exchange Upd.req( Diag Flag, Inp Data ) | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 92 | DATA-EXCH | MSCY1S Set Cfg.req ( Cfg Data )<br>=><br>Diag.Cfg_Fault:=TRUE<br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Get Cfg Upd.req ( Cfg Data )<br>MSCY1S Set Cfg.cnf(+) ()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 93 | DATA-EXCH | MSCY1S Get Output.req ()<br>/DX_Entered = TRUE<br>=><br>Clear Flag := Safe_State<br>New Flag := New Output<br>New Output := FALSE<br>Outp Data, Output Data := B Output<br>DMPMS RD Outp Upd.req ( Outp Data )<br>MSCY1S Get Output.cnf(+) ( Output Data, Clear Flag, New Flag ) | DATA-EXCH |
| 94 | DATA-EXCH | MSCY1S Get Output.req ()<br>/DX_Entered = FALSE<br>=><br>MSCY1S Get Output.cnf(-) () | DATA-EXCH |
| 95 | DATA-EXCH | MSCY1S Set Input.req ( Input Data )<br>/Diag.Freeze_Mode = FALSE<br>=><br>Inp Data, B Input := Input Data<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>MSCY1S Set Input.cnf(+) () | DATA-EXCH |
| 96 | DATA-EXCH | MSCY1S Set Input.req ( Input Data )<br>/Diag.Freeze_Mode = TRUE<br>=><br>B Freeze := Input Data<br>MSCY1S Set Input.cnf(+) () | DATA-EXCH |
| 97 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | DATA-EXCH |
| 98 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 2<br>=><br>Status := New Prm<br>Prm_Pending := 1<br>User Parameter Data := B User Prm<br>MSCY1S Check User Prm Result.cnf(-) ( Status )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | DATA-EXCH |
| 99 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add # Check_Prm_Add<br>=><br>Status := Inv Master Add<br>Prm_Pending := 0<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | DATA-EXCH |
| 100 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Prm_OK = TRUE<br>=><br>Prm_Pending := 0<br>MSCY1S Check User Prm Result.cnf(+)() | DATA-EXCH |
| 101 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Prm_OK = FALSE<br>=><br>Diag.Prm_Fault := TRUE<br>LEAVE-MASTER<br>Prm_Pending := 0<br>Act_Ref := Act_Ref + 1<br>MSCY1S Check User Prm Result.cnf(+)()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 102 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | DATA-EXCH |
| 103 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 2<br>=><br>Status := New Cfg<br>Cfg_Pending := 1<br>Cfg Data := B Cfg<br>MSCY1S Check Cfg Result.cnf(-) ( Status )<br>MSCY1S Check Cfg.ind ( Cfg Data ) | DATA-EXCH |
| 104 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add # Check_Cfg_Add<br>=><br>Status := Inv Master Add<br>Cfg_Pending := 0<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | DATA-EXCH |
| 105 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Input Data Len = Inp Data Len<br>&& Output Data Len = Outp Data Len<br>=><br>Cfg_Pending := 0<br>MSCY1S Check Cfg Result.cnf(+) () | DATA-EXCH |
| 106 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Input Data Len <> Inp Data Len<br>&& Output Data Len <> Outp Data Len<br>=><br>Diag.Cfg_Fault := TRUE<br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>MSCY1S Check Cfg Result.cnf(+) ()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 107 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = FALSE<br>=><br>Diag.Cfg_Fault := TRUE<br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>MSCY1S Check Cfg Result.cnf(+) ()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 108 | DATA-EXCH | MSCY1S Application Ready.req()<br>/Diag.Stat_Diag = TRUE<br>=><br>Diag.Stat_Diag := FALSE<br>Diag_Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Data Exchange Upd.req( Diag Flag, Inp Data ) | DATA-EXCH |
| 109 | DATA-EXCH | MSCY1S Application Ready.req()<br>/Diag.Stat_Diag = FALSE<br>=><br>ignore | DATA-EXCH |
| 110 | DATA-EXCH | MSAC1S Abort.ind()<br>=><br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 111 | DATA-EXCH | MSAC1S Start.cnf()<br>/Start_State <> B<br>=><br>START_C1_CON | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 112 | DATA-EXCH | MSAC1S Start.cnf()<br>/Start_State = B<br>=><br>START_C1_CON<br>MSCY1S_Start.ind ( Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | DATA-EXCH |
| 113 | DATA-EXCH | MSAC1S Stop.cnf()<br>/Start_State <> E<br>=><br>STOP_C1_CON | DATA-EXCH |
| 114 | DATA-EXCH | MSAC1S Stop.cnf()<br>/Start_State = E<br>=><br>STOP_C1_CON<br>MSCY1S_Stop.ind | DATA-EXCH |
| 115 | DATA-EXCH | DMPMS Set Slave Add.ind(New Slave Add, Ident Number, No Add Chg, Rem Slave Data)<br>=><br>ignore | DATA-EXCH |
| 116 | DATA-EXCH | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add = Req_Add &&<br>( Diag.Stat_Diag = TRUE \|\| Reference < Ref_Cnt)<br>=><br>TRIG_WD | DATA-EXCH |
| 117 | DATA-EXCH | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add = Req_Add && Diag.Stat_Diag = FALSE && Reference >= Ref_Cnt<br>=><br>Diag Flag := FALSE<br>TRIG_WD<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>MSCY1S Set Slave Diag.cnf(+) (Reference := Act_Cnt ) | DATA-EXCH |
| 118 | DATA-EXCH | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | DATA-EXCH |
| 119 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = FALSE && Unlock_Req = FALSE<br>=><br>DMPMS Set minTsdr.req ( MinTsdr ) | DATA-EXCH |
| 120 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE && Diag.Master_Add <> Req_Add<br>=><br>ignore | DATA-EXCH |
| 121 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE && Diag.Master_Add = Req_Add<br>=><br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 122 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE<br>&& Diag.Master_Add = Req_Add && (Ident_Number <> Diag.Ident_Number<br>\|\| !OPERATION_MODE_OK \|\| WD_On && (WD_Fact_1=0 \|\| WD_Fact_2=0))<br>=><br>Diag.Prm_Fault := TRUE<br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 123 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE<br>&& Diag.Master_Add = Req_Add && Ident_Number = Diag.Ident_Number<br>&& OPERATION_MODE_OK && !WD_On \|\| (WD_Fact_1>0 && WD_Fact_2>0)<br>&& (Freeze_Req && Freeze Supp = FALSE<br>\|\| Sync_Req && Sync Supp = FALSE<br>\|\| Prm_Data[1].0, .1, .2 = TRUE)<br>=><br>Diag.Not_Supported := TRUE<br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 124 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE<br>&& Diag.Master_Add <> Req_Add && (Ident_Number <> Diag.Ident_Number<br>\|\| !OPERATION_MODE_OK \|\| WD_On && (WD_Fact_1=0 \|\| WD_Fact_2=0)<br>\|\| Freeze_Req && !Freeze Supp<br>\|\| Sync_Req && !Sync Supp<br>\|\| Prm_Data[1].0, .1, .2 = TRUE)<br>=><br>ignore | DATA-EXCH |
| 125 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK &&!PV0V0 && NOPRMCMD<br>=><br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 126 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK &&<br>!NOPRMCMD && Prm_Pending = 0<br>=><br>SET_WD<br>Active_Groups := Group_Ident<br>EXE_PRM_CMD<br>Prm_Pending := 1<br>Check_Prm_Add := Req Add<br>User Parameter Data := Prm Data.User Prm<br>Diag_Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | DATA-EXCH |
| 127 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK &&<br>!NOPRMCMD && Prm_Pending > 0<br>=><br>SET_WD<br>Active_Groups := Group_Ident<br>EXE_PRM_CMD<br>Prm_Pending := 2<br>Check_Prm_Add := Req Add<br>B User Prm := Prm Data.User Prm<br>Diag_Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 128 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK && NOPRMCMD && PV0V0 &&<br>Prm_Pending = 0<br>=><br>SET_WD<br>Active_Groups := Group_Ident<br>Prm_Pending := 1<br>Check_Prm_Add := Req Add<br>User Parameter Data := Prm Data.User Prm<br>Diag_Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | DATA-EXCH |
| 129 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK && NOPRMCMD && PV0V0 &&<br>Prm_Pending > 0<br>=>SET_WD<br>Active_Groups := Group_Iden<br>tPrm_Pending := 2<br>Check_Prm_Add := Req Add<br>B User Prm := Prm Data.User Prm<br>Diag_Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | DATA-EXCH |
| 130 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add <> Req_Add && PRM_OK && Prm_Pending = 0<br>=><br>Diag.Sync_Mode := FALSE<br>Diag.Freeze_Mode := FALSE<br>Diag.Station_Not_Ready := TRUE<br>Diag.Stat_Diag := FALSE<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups := Group_Ident<br>Safe_State := TRUE<br>Diag.Master_Add := Req_Add<br>Check_Prm_Add := Req Add<br>STOP_ASM<br>STOP_C1<br>Prm_Pending := 1<br>User Parameter Data := Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Diag_Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Leave.req<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 131 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add <> Req_Add && PRM_OK && Prm_Pending > 0<br>=><br>Diag.Sync_Mode := FALSE<br>Diag.Freeze_Mode := FALSE<br>Diag.Station_Not_Ready := TRUE<br>Diag.Stat_Diag := FALSE<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups := Group_Ident<br>Safe_State := TRUE<br>Diag.Master_Add := Req_Add<br>Check_Prm_Add := Req Add<br>STOP_ASM<br>STOP_C1<br>Prm_Pending := 2<br>B User Prm := Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Leave.req<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | WAIT-CFG |
| 132 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len < 7<br>=><br>ignore | DATA-EXCH |
| 133 | DATA-EXCH | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | DATA-EXCH |
| 134 | DATA-EXCH | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add = Req_Add && Cfg_Pending = 0<br>=><br>Cfg_Pending := 1<br>Check_Cfg_Add := Req Add<br>TRIG_WD<br>MSCY1S Check Cfg.ind ( Check Cfg Mode, Cfg Data ) | DATA-EXCH |
| 135 | DATA-EXCH | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add = Req_Add && Cfg_Pending > 0<br>=><br>Cfg_Pending := 2<br>Check_Cfg_Add := Req Add<br>B Cfg := Cfg Data<br>TRIG_WD | DATA-EXCH |
| 136 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len > 0 && Outp_Data.len = Outp Data Len<br>&& Diag.Sync_Mode = FALSE<br>=><br>B Output := Outp Data<br>TRIG_WD<br>Safe_State := FALSE<br>Clear Flag := FALSE<br>New Output := TRUE<br>CHECK_ALARM_START<br>MSCY1S New Output.ind ( Clear Flag ) | DATA-EXCH |
| 137 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len > 0 && Outp_Data.len = Outp Data Len<br>&& Diag.Sync_Mode = TRUE<br>=><br>B Sync := Outp Data<br>TRIG_WD<br>Safe_State := FALSE<br>CHECK_ALARM_START | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 138 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len = 0 && Outp Data Len = 0<br>=><br>TRIG_WD<br>CHECK_ALARM_START | DATA-EXCH |
| 139 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len <> Outp Data Len && (Outp_Data.len = 0 && Fail_Safe_supp)<br>=><br>TRIG_WD<br>Safe_State := TRUE<br>Clear Flag := TRUE<br>New Output := TRUE<br>CHECK_ALARM_START<br>MSCY1S New Output.ind ( Clear Flag ) | DATA-EXCH |
| 140 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len <> Outp Data Len &&<br>(Outp_Data.len <> 0 \|\| Fail_Safe_supp )<br>=><br>LEAVE-MASTERAct_Ref := Act_Ref + 1<br>Diag.Cfg_Fault := TRUE<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 141 | DATA-EXCH | WDTimer expired<br>=><br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 142 | DATA-EXCH | DMPMS Global Control.ind(Control Command, Group Select)<br>/ (Control_Command.0, .6, .7 = TRUE)<br>=><br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 143 | DATA-EXCH | DMPMS Global Control.ind(Control Command, Group Select)<br>/Group_Select <> 0 && ((Active_Groups & Group_Select) = 0) && ((Group_Select < 0x80 ) \|\| Isochronous Mode = FALSE) && (Control_Command.0, .6, .7 = FALSE)<br>=><br>ignore | DATA-EXCH |
| 144 | DATA-EXCH | DMPMS Global Control.ind(Control Command, Group Select)<br>/Control Command.Clear Command = FALSE<br>&& (Group_Select = 0 \|\| (Active_Groups & Group_Select) <> 0<br>\|\| ((Group_Select >= 0x80 ) && Isochronous Mode = TRUE))<br>&& (Control_Command.0, .6, .7 = FALSE)<br>=><br>Clear Command := FALSE | CHECK-ISOM |
| 145 | DATA-EXCH | DMPMS Global Control.ind(Control Command, Group Select)<br>/Control Command.Clear Command = TRUE<br>&& (Group_Select = 0 \|\| (Active_Groups & Group_Select) <> 0<br>\|\| ((Group_Select >= 0x80 ) && Isochronous Mode = TRUE))<br>&& (Control_Command.0, .6, .7 = FALSE)<br>=><br>Clear Command := TRUE<br>Clear Flag := TRUE<br>New Output := TRUE<br>Safe_State := TRUE | CHECK-ISOM |
| 146 | DATA-EXCH | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add = Req_Add<br>=><br>Diag.Prm_Fault := TRUE<br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 147 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 0<br>=><br>Status := Not pending<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | DATA-EXCH |
| 148 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 2<br>=><br>Status := New Prm<br>Ext_Prm_Pending := 1<br>Ext_User Parameter Data := B Ext User Prm<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status )<br>MSCY1S Check Ext User Prm.ind ( Ext User Parameter Data ) | DATA-EXCH |
| 149 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add <> Check_Ext_Prm_Add<br>=><br>Status := Inv Master Add<br>Ext_Prm_Pending := 0<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | DATA-EXCH |
| 150 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add = Check_Ext_Prm_Add &&<br>Ext_Prm_OK = TRUE<br>=><br>Ext_Prm_Pending := 0<br>MSCY1S Check Ext User Prm Result.cnf(+)() | DATA-EXCH |
| 151 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add = Check_Ext_Prm_Add &&<br>Ext_Prm_OK = FALSE<br>=><br>Diag.Cfg_Fault := TRUE<br>LEAVE-MASTER<br>Ext_Prm_Pending := 0<br>Act_Ref := Act_Ref + 1<br>MSCY1S Check Ext User Prm Result.cnf(+)()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 152 | CHECK-SYNC | /NOSYNC<br>=><br>Sync Command := 0 | CHECK-FREEZE |
| 153 | CHECK-SYNC | /UNSYNC && Sync Supp = FALSE<br>=><br>Sync Command := 2 | CHECK-FREEZE |
| 154 | CHECK-SYNC | /SYNC && Sync Supp = FALSE<br>=><br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 155 | CHECK-SYNC | /UNSYNC && Sync Supp = TRUE && B Sync <> Nil<br>=><br>B Output := B Sync<br>B Sync := Nil<br>Sync Command := 2<br>Clear Flag := Safe_State<br>New Outpu t := TRUE<br>Diag.Sync_Mode := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>MSCY1S New Output.ind ( Clear Flag )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | CHECK-FREEZE |
| 156 | CHECK-SYNC | /UNSYNC && Sync Supp = TRUE && B Sync = Nil<br>=><br>Sync Command := 2<br>Clear Flag := Safe_State<br>Diag.Sync_Mode := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | CHECK-FREEZE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 157 | CHECK-SYNC | /SYNC && Sync Supp = TRUE && Diag.Sync_Mode = FALSE<br>=><br>B Sync := Nil<br>Sync Command := 1<br>Diag.Sync_Mode := TRUE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | CHECK-FREEZE |
| 158 | CHECK-SYNC | /SYNC && Sync Supp = TRUE && Diag.Sync_Mode = TRUE && B Sync <> Nil<br>=><br>B Output := B Sync<br>B Sync := Nil<br>Clear Flag := Safe State<br>New Output := TRUE<br>Sync Command := 1<br>MSCY1S New Output.ind ( Clear Flag ) | CHECK-FREEZE |
| 159 | CHECK-SYNC | /SYNC && Sync Supp = TRUE && Diag.Sync_Mode = TRUE && B Sync = Nil<br>=><br>Sync Command := 1 | CHECK-FREEZE |
| 160 | CHECK-ISOM | /(Group_Select >= 0x80 ) && Isochronous Mode = TRUE<br>=><br>if (FirstSynch = TRUE)<br>Status:= IsoM Start<br>FirstSynch:= FALSE<br>else<br>Status:=IsoM SYNCH<br>endif<br>MSCY1S SYNCH Event.ind(Status) | CHECK-SYNC |
| 161 | CHECK-ISOM | /(Group_Select < 0x80 ) \|\| Isochronous Mode = FALSE<br>=> | CHECK-SYNC |
| 162 | CHECK-FREEZE | /NOFREEZE<br>=><br>Freeze Command := 0<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 163 | CHECK-FREEZE | /UNFREEZE && Freeze Supp = FALSE<br>=><br>Freeze Command := 2<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 164 | CHECK-FREEZE | /FREEZE && Freeze Supp = FALSE<br>=><br>LEAVE-MASTER<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref ) | WAIT-PRM |
| 165 | CHECK-FREEZE | /UNFREEZE && Freeze Supp = TRUE && B Freeze <> Nil<br>=><br>Inp Data := B Freeze<br>B Freeze := Nil<br>Freeze Command := 2<br>Diag.Freeze_Mode := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 166 | CHECK-FREEZE | /UNFREEZE && Freeze Supp = TRUE && B Freeze = Nil<br>=><br>Freeze Command := 2<br>Diag.Freeze_Mode := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 167 | CHECK-FREEZE | /FREEZE && Freeze Supp = TRUE && Diag.Freeze_Mode = FALSE<br>=><br>B Freeze := Nil<br>Freeze Command := 1<br>Diag.Freeze_Mode := FALSE<br>Diag Data := Diag<br>Act_Ref := Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference := Act_Ref )<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 168 | CHECK-FREEZE | /FREEZE && Freeze Supp = TRUE && Diag.Freeze_Mode = TRUE && B Freeze <> Nil<br>=><br>Inp Data := B Freeze<br>B Freeze := Nil<br>Freeze Command := 1<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 169 | CHECK-FREEZE | /FREEZE && Freeze Supp = TRUE && Diag.Freeze_Mode = TRUE && B Freeze = Nil<br>=><br>Freeze Command := 1<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 170 | any state | MSCY1S Reset.req ()<br>=><br>DMPMS Slave Deact.req () | SL-DEACT |
| 171 | any state | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | SAME |
| 172 | SL-DEACT | DMPMS Slave Deact.cnf ()<br>=><br>MSCY1S Reset.cnf () | |

### 9.1.4 Functions

Table 49 contains the functions used by the MSCY1S and their descriptions.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 49 – Functions used by the MSCY1S**

| Name | Function |
|---|---|
| LEAVE-MASTER | Diag.Master_Add=invalid<br>Diag.Sync_Mode=FALSE<br>Diag.Freeze_Mode=FALSE<br>Diag.Prm_Req=TRUE<br>Diag.Station_Not_Ready=TRUE<br>Stop Timer(WD), Diag.WD_On=FALSE<br>Diag.Stat_Diag=FALSE, Operation_Mode=V0<br>Safe_State=TRUE<br>STOP_ASM,<br>STOP_C1,<br>STOP_ISOM<br>DMPMS_Leave.Req<br>Diag_Data=Diag |
| PRM_OK | Prm_Data.len>=7<br>&& Lock_Req=TRUE<br>&& Unlock_Req=FALSE<br>&& Ident_Number=Diag.Ident_Number<br>&& (WD_On=FALSE ||<br>(WD_On=TRUE && WD_Fact_1>0 && WD_Fact_2>0))<br>&& (Freeze_Req=FALSE || Freeze_Supported)<br>&& (Sync_Req=FALSE || Sync_Supported)<br>&& (Prm_Data[1].0, .1, .2=FALSE)<br>&& OPERATION_MODE_OK |
| SET_WD | if (WD_On=TRUE)<br>Start Timer(WD),<br>Diag.WD_On=TRUE)<br>else<br>Stop Timer(WD),<br>Diag.WD_On=FALSE<br>endif |
| TRIG_WD | if (Diag.WD_On=TRUE)<br>Start Timer(WD)<br>endif |
| NOSYNC | Control_Command.Sync=0 && Control_Command.UnSync=0 |
| SYNC | Control_Command.Sync=1 && Control_Command.UnSync=0 |
| UNSYNC | Control_Command.UnSync=1 |
| NOFREEZE | Control_Command.Freeze=0 && Control_Command.UnFreeze=0 |
| FREEZE | Control_Command.Freeze=1 && Control_Command.UnFreeze=0 |
| UNFREEZE | Control_Command.UnFreeze=1 |
| STOP_ISOM | if(Isochronous Mode Supp = TRUE<br>&& Isochronous Mode =TRUE<br>&& First Synch = FALSE)<br>MSCY1S SYNCH Event.ind(Status:= IsoM Stop)<br>First Synch = TRUE<br>endif |
| STOP_ASM | if (DX_Entered=TRUE)<br>MSCY1S_Stop.ind<br>endif |
| CHECK_ALARM_START | if (DX_Entered=FALSE && Diag.Stat_Diag=FALSE)<br>DX_Entered=TRUE,<br>MSCY1S_Start.ind(Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit)<br>endif |
| SET_OPERATION_MODE | if (Prm_Data.len >= 10 && DPV1_Enable=TRUE)<br>Operation_Mode=V1<br>else<br>Operation_Mode=V0<br>endif<br>if (Prm_Data.len >= 10 && IsoM_Req =TRUE)<br>Isochron_Mode=True<br>else<br>Isochron_Mode=FALSE<br>endif |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Name | Function |
|---|---|
| PV0V0 | Operation_Mode = V0<br>&& (Prm_Data.len < 10 \|\| DPV1_Enable = FALSE) |
| OPERATION_MODE_OK | if (IsARexistent(MS1) && DPV1_Enable )<br>further checks according the table “Rules for DPV1_Status_1, DPV1_Status_2, and DPV1_Status_3 check” |
| NOPRMCMD | !(DPV1_Status_3.PrmCmd = 1 && Prm_Data.len = 18 && PrmCmd_supp) |
| SET_ALARM_CHKCFGM | if (Prm_Data.len >=10 && DPV1_Enable=TRUE)<br>Enabled_Alarms=DPV1_Status_2.2 - DPV1_Status_2.7<br>else<br>Enabled_Alarms=0<br>endif<br>if (Prm_Data.len >=10)<br>Alarm_Mode_Master=DPV1_Status_3.0 - DPV1_Status_3.2,<br>Chk Cfg Mode = DPV1_Status_2.0,<br>Prm Structure = DPV1_Status_3.4,<br>if (DPV1_Status3.PrmCmd&& Prm_Data.PrmCmd exist)<br>EXE_PRM_CMD<br>else<br>Alarm_Mode_Master=0,<br>Chk Cfg Mode = FALSE,<br>Prm Structure = FALSE<br>endif |
| EXE_PRM_CMD | if (PrmCmd.Function.Primary)<br>Primary=TRUE,<br>if(Start_State=B\|\|Start_State=BEB\|\|Start_State=EB\|\|Start_State=Run)<br>STOP_C1, START_C1<br>else<br>START_C1<br>endif<br>else<br>Primary=FALSE<br>endif<br>if (PrmCmd.Function.Stop_MS1)<br>if(Start_State=B\|\|Start_State=BEB\|\|Start_State=EB\|\|Start_State=Run)<br>STOP_C1<br>endif<br>endif<br>if (PrmCmd.Function.Start_MS1)<br>if(Start_State=E\|\|Start_State=BE\|\|Start_State=Idle)<br>START_C1<br>endif<br>endif |
| START_C1 | if (Start_State=Idle)<br>MSAC1S_Start.req(Diag.Master_Add),<br>Start_State=B<br>endif<br>if (Start_State=E)<br>Start_State=EB<br>endif<br>if (Start_State=BE)<br>Start_State=BEB<br>endif |
| STOP_C1 | if (Start_State=B)<br>Start_State=BE<br>endif<br>if (Start_State=Run)<br>MSAC1S_Stop.req,<br>Start_State=E<br>endif<br>if (Start_State=EB)<br>Start_State=E<br>endif<br>if (Start_State=BEB)<br>Start_State=BE<br>endif |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Name | Function |
|---|---|
| START_C1_CON | if (Start_State=B)<br>  Start_State=Run<br>endif<br>if (Start_State=BE)<br>  MSAC1S_Stop.req, Start_State=E<br>endif<br>if (Start_State=BEB)<br>  MSAC1S_Stop.req, Start_State=EB<br>endif |
| STOP_C1_CON | if (Start_State=E)<br>  Start_State=Idle<br>endif<br>if (Start_State=EB)<br>  MSAC1S_Start.req(Diag.Master_Add),<br>  Start_State=B<br>endif |
| LEAVE-MASTER | Diag.Master_Add=invalid<br>Diag.Sync_Mode=FALSE<br>Diag.Freeze_Mode=FALSE<br>Diag.Prm_Req=TRUE<br>Diag.Station_Not_Ready=TRUE<br>Stop Timer(WD), Diag.WD_On=FALSE<br>Diag.Stat_Diag=FALSE, Operation_Mode=V0<br>Safe_State=TRUE<br>STOP_ASM,<br>STOP_C1,<br>STOP_ISOM<br>DMPMS_Leave.Req<br>Diag_Data=Diag |

## 9.2 MSAC1S

### 9.2.1 Primitive definitions

#### 9.2.1.1 Primitives exchanged between MSAC1S and FSPMS

Table 50 shows the service primitives including their associated parameters issued by the FSPMS and received by the MSAC1S.

**Table 50 – Primitives issued by FSPMS to MSAC1S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MS1.req | FSPMS | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMS | (none) | |
| Read.rsp(+) | FSPMS | Length,<br>Data | |
| Read.rsp(-) | FSPMS | Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.rsp(+) | FSPMS | Length | |
| Write.rsp(-) | FSPMS | Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Alarm Ack.rsp(+) | FSPMS | Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Alarm Ack.rsp(-) | FSPMS | (none) | |

Table 51 shows the service primitives including their associated parameters issued by MSAC1S and received by the FSPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 51 – Primitives issued by MSAC1S to FSPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MS1.cnf | MSAC1S | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | MSAC1S | (none) | |
| Fault.ind | MSAC1S | (none) | |
| Read.ind | MSAC1S | Slot Number,<br>Index,<br>Length | |
| Write.ind | MSAC1S | Slot Number,<br>Index,<br>Length,<br>Data | |
| Alarm Ack.ind | MSAC1S | Slot Number,<br>Alarm Type,<br>Seq Nr | |

#### 9.2.1.2 Primitives exchanged between MSAC1S and MSCY1S

Table 52 shows the service primitives including their associated parameters issued by MSCY1S and received by the MSAC1S.

**Table 52 – Primitives issued by MSCY1S to MSAC1S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Start.req | MSCY1S | Master Add | Start Processing of acyclic services |
| Stop.req | MSCY1S | (none) | Stop Processing of acyclic services |

Table 53 shows the service primitives including their associated parameters issued by the MSAC1S and received by the MSCY1S.

**Table 53 – Primitives issued by MSAC1S to MSCY1S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Start.cnf | MSAC1S | (none) | MSAC1 is ready to execute acyclic services |
| Stop.cnf | MSAC1S | (none) | MSAC1 is disabled |
| Abort.ind | MSAC1S | (none) | Termination of MSAC1 due to a sequence error or wrong protocol elements |

#### 9.2.1.3 Parameter of MSAC1S primitives

The parameters used with the primitives exchanged between the FSPMS and the MSAC1S are described in FAL Service Definition in IEC 61158-5-3. Additional parameters are described in Table 54.

**Table 54 – Parameter used with primitives exchanged between MSAC1S and MSCY1S**

| Parameter name | Description |
|---|---|
| MasterAdd | This parameter conveys the DL address of the assigned DP-master. |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 9.2.2 State machine description

After Power-On the MSAC1S waits for initiation by means of SInit MS1.

In the state CLOSED the service Start called from the MSCY1S is expected. Then the access protection for the SAP 51 (50) will be activated by means of the function RSAP_ACTIVATE.

When receiving the confirmation the State Machine enters the OPEN state. In this state the services Read, Write and Alarm_Ack are allowed. Only one request from the DP-master is processed simultaneously on each SAP. Additionally to Read/Write at SAP 51 a Alarm_Ack may be executed at SAP 50.

When a valid service is invoked by the DP-master, the MSAC1S will indicate this primitive to the AP-Context (Read/Write or Alarm_Ack) and waits for response. After receiving the corresponding response MSAC1S will provide this PDU in the corresponding SAP update-buffer. The service is finished after DP-master has fetched the response.

The local service Stop called from the MSCY1S is closing the connection. The SAP 51 (50) is deactivated and a possibly stored response in the DL is deleted.

The abort of the connection is indicated to the MSCY1S with the local service Abort. If the User is still processing a response while the connection is aborted this response shall be discarded by the User.

**Local variables of the MSAC1S**

**Server_SAP**
(Unsigned8)

This SAP (51) is used for the MSAC1S_Read/_Write and MSAC1S_Alarm_Ack services.

**Alarm_SAP**
(Unsigned8)

This SAP (50) is reserved for MSAC1S_Alarm_Ack services.

**Res_SAP**
(Unsigned8)

Variable holding the SAP for the MSAC1S_Alarm_Ack -Response.

**Service_Header**
(Unsigned8)

The actual service header of the service processed. Used to check whether the User generates the correct response.

**AA_State**
MSAC1S_Alarm_Ack services may be executed simultaneousely with the MSAC1S_Read-/_Write-services. Thus, the actual state of the MSAC1S_Alarm_Ack processing is stored in the variable AA_State:

Values of AA_State:

| | |
|---|---|
| Idle | No MSAC1S_Alarm_Ack in progress |
| WRes | MSAC1S_Alarm_Ack.ind is sent to the MSAL1S waiting for response |
| Upd | Response of the MSAL1S will be sent to the Update-Buffer |
| SRes | Response is stored – waiting for the next poll of the DP-master |

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**VS_Upd**
Indicates that a Response will be sent to the Update-Buffer.

### 9.2.3 MSAC1S state table

Table 55 contains the complete description of the MSAC1S state machine.

**Table 55 – MSAC1S state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | MSAC1S_SInit_MS1.req<br>=><br>Server_SAP:=51<br>Alarm_SAP:=50<br>MSAC1S_SInit_MS1.cnf | CLOSED |
| 2 | POWER-ON | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 3 | POWER-ON | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 4 | POWER-ON | MSAC1S_Alarm_Ack.rsp(-) ()<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 5 | CLOSED | MSAC1S_Start.req(Master_Add)<br>=><br>DMPMS_RSAP_ACTIVATE.req(SSAP=Server_SAP, Access=Master_Add, L_sdu_length_list=Nil,Indication_Mode=Unchanged) | SET-ACC51 |
| 6 | CLOSED | MSAC1S_Stop.req<br>=><br>MSAC1S_Stop.cnf | CLOSED |
| 7 | SET-ACC51 | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status = OK<br>=><br>DMPMS_RSAP_ACTIVATE.req(SSAP=Alarm_SAP, Access=Master_Add, L_sdu_length_list=Nil,Indication_Mode=Unchanged) | SET-ACC50 |
| 8 | SET-ACC51 | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 9 | SET-ACC50 | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Alarm_SAP,M_status)<br>/M_status = OK<br>=><br>AA_State:=Idle<br>VS_Upd:=False<br>MSAC1S_Start.cnf | OPEN |
| 10 | SET-ACC50 | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Alarm_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 11 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 12 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 13 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Read-REQ-PDU( )<br>=><br>Service_Header:=Function_Num<br>MSAC1S_Read.ind(Req_Add=Loc_add, Slot_Number,Index,Length) | VS-WRES |
| 14 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Write-REQ-PDU( ) && L_sdu.len >= (Length+4)<br>=><br>Service_Header:=Function_Num<br>MSAC1S_Write.ind(Req_Add=Loc_add, Slot_Number,Index,Length,Data) | VS-WRES |
| 15 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Alarm_Ack-REQ-PDU( )<br>=><br>AA_State:=WRes<br>Res_SAP:=Server_SAP<br>MSAC1S_Alarm_Ack.ind(Alarm_Type, Slot_Number,Seq_Nr) | AA-WRES |
| 16 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Alarm_Ack-REQ-PDU( )<br>=><br>AA_State:=WRes<br>Res_SAP:=Alarm_SAP<br>MSAC1S_Alarm_Ack.ind(Alarm_Type, Slot_Number,Seq_Nr) | AA-WRES |
| 17 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT(Read-REQ-PDU( ) \|\| Write-REQ-PDU( ) && (L_sdu.len >= Length+4) \|\| Alarm_Ack-REQ-PDU( ))<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 18 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT(Alarm_Ack-REQ-PDU( ))<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 19 | OPEN | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 20 | OPEN | MSAC1S_Read.rsp(+)(Length, Data)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 21 | OPEN | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 22 | OPEN | MSAC1S_Write.rsp(+)(Length)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 23 | OPEN | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 24 | OPEN | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 25 | OPEN | MSAC1S_Alarm_Ack.rsp(-) ()<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 26 | OPEN | MSAC1S_Stop.req<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | STOP-DEA51 |
| 27 | OPEN | MSAC1S_Start.req(Master_Add)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 28 | OPEN | MSAC1S_SInit_MS1.req<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 29 | OPEN | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 30 | VS-WRES | MSAC1S_Read.rsp(+)(Length, Data)<br>/Service_Header=0x5E && Max_DLSDU_Length_Req_Low>=(Length+4)<br>=><br>VS_Upd:=True<br>L_sdu:=Read-RES-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu,<br>Serv_class=Low,Transmit=Single) | VS-SRES |
| 31 | VS-WRES | MSAC1S_Read.rsp(+)(Length, Data)<br>/Service_Header=0x5E && Max_DLSDU_Length_Req_Low<(Length+4)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 32 | VS-WRES | MSAC1S_Write.rsp(+)(Length)<br>/Service_Header=0x5F<br>=><br>VS_Upd:=True<br>L_sdu:=Write-RES-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu,<br>Serv_class=Low,Transmit=Single) | VS-SRES |
| 33 | VS-WRES | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>/Service_Header=0x5E<br>=><br>VS_Upd:=True<br>L_sdu:=Read-NRS-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu,<br>Serv_class=Low,Transmit=Single) | VS-SRES |
| 34 | VS-WRES | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>/Service_Header=0x5F<br>=><br>VS_Upd:=True<br>L_sdu:=Write-NRS-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu,<br>Serv_class=Low,Transmit=Single) | VS-SRES |
| 35 | VS-WRES | MSAC1S_Read.rsp(+)(Length, Data)<br>/Service_Header<>0x5E<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 36 | VS-WRES | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>/Service_Header<>0x5E<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 37 | VS-WRES | MSAC1S_Write.rsp(+)(Length)<br>/Service_Header<>0x5F<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 38 | VS-WRES | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>/Service_Header<>0x5F<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 39 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 40 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 41 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 42 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Alarm_Ack-REQ-PDU( ) && AA_State=Idle<br>=><br>AA_State:=WRes<br>Res_SAP:=Alarm_SAP<br>MSAC1S_Alarm_Ack.ind(Alarm_Type, Slot_Number,Seq_Nr) | VS-WRES |
| 43 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=SRes<br>=><br>AA_State:=Idle | VS-WRES |
| 44 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT((Alarm_Ack-REQ-PDU( ) && AA_State=Idle)<br>\|\| (Update_status=LO && L_sdu.len=0 && AA_State=SRes))<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 45 | VS-WRES | MSAC1S_Start.req(Master_Add)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 46 | VS-WRES | MSAC1S_Stop.req<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | STOP-DEA51 |
| 47 | VS-WRES | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>/AA_State<>WRes<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 48 | VS-WRES | MSAC1S_Alarm_Ack.rsp(-) ()<br>/AA_State<>WRes<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 49 | VS-WRES | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>/AA_State=WRes<br>=><br>AA_State:=Upd<br>DMPMS_REPLY_UPDATE.req(SSAP=Res_SAP, L_sdu=Alarm_Ack-RES-PDU, Serv_class=Low,Transmit=Single) | VS-WRES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 50 | VS-WRES | MSAC1S_Alarm_Ack.rsp(-) ()<br>/AA_State=WRes<br>=><br>AA_State:=Upd<br>DMPMS_REPLY_UPDATE.req(SSAP=Res_SAP, L_sdu=Alarm_Ack-NRS-PDU, Serv_class=Low,Transmit=Single) | VS-WRES |
| 51 | VS-WRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Alarm_SAP,Serv_class,L_status)<br>/L-status=OK && AA_State=Upd<br>=><br>AA_State:=SRes | VS-WRES |
| 52 | VS-WRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Alarm_SAP,Serv_class,L_status)<br>/L-status=OK && AA_State<>Upd<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 53 | VS-WRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L_status=LS/IV/LR<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 54 | VS-WRES | MSAC1S_SInit_MS1.req<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 55 | VS-WRES | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 56 | VS-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L-status=OK && VS_Upd=True<br>=><br>VS_Upd:=False | VS-SRES |
| 57 | VS-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L_status=LS/IV/LR<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 58 | VS-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L-status=OK && VS_Upd=False<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 59 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 60 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 61 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=Idle&& VS_Upd=False | OPEN |
| 62 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=WRes && VS_Upd=False | AA-WRES |
| 63 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && (AA_State=SRes\|\| AA_State=Upd) && VS_Upd=False | AA-SRES |
| 64 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && VS_Upd=True<br>=><br>MSAC1S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 65 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len<>0<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 66 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Alarm_Ack-REQ-PDU( ) && AA_State=Idle<br>=><br>AA_State:=WRes<br>Res_SAP:=Alarm_SAP<br>MSAC1S_Alarm_Ack.ind(Alarm_Type, Slot_Number,Seq_Nr) | VS-SRES |
| 67 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=SRes<br>=><br>AA_State:=Idle | VS-SRES |
| 68 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT((Alarm_Ack-REQ-PDU( ) && AA_State=Idle)<br>\|\| (Update_status=LO && L_sdu.len=0 && AA_State=SRes ))<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 69 | VS-SRES | MSAC1S_Start.req(Master_Add)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 70 | VS-SRES | MSAC1S_Stop.req<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | STOP-DEA51 |
| 71 | VS-SRES | MSAC1S_Read.rsp(+)(Length, Data)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 72 | VS-SRES | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 73 | VS-SRES | MSAC1S_Write.rsp(+)(Length)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 74 | VS-SRES | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 75 | VS-SRES | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>/AA_State<>WRes<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 76 | VS-SRES | MSAC1S_Alarm_Ack.rsp(-) ()<br>/AA_State<>WRes<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 77 | VS-SRES | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>/AA_State=WRes<br>=><br>AA_State:=Upd<br>DMPMS_REPLY_UPDATE.req(SSAP=Res_SAP, L_sdu=Alarm_Ack-RES-PDU,<br>Serv_class=Low,Transmit=Single) | VS-SRES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 78 | VS-SRES | MSAC1S_Alarm_Ack.rsp(-) ()<br>/AA_State=WRes<br>=><br>AA_State:=Upd<br>DMPMS_REPLY_UPDATE.req(SSAP=Res_SAP, L_sdu=Alarm_Ack-NRS-PDU, Serv_class=Low,Transmit=Single) | VS-SRES |
| 79 | VS-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Alarm_SAP,Serv_class,L_status)<br>/L-status=OK && AA_State=Upd<br>=><br>AA_State:=SRes | VS-SRES |
| 80 | VS-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Alarm_SAP,Serv_class,L_status)<br>/L-status=OK && AA_State<>Upd<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 81 | VS-SRES | MSAC1S_SInit_MS1.req<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 82 | VS-SRES | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 83 | AA-WRES | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>=><br>AA_State:=Upd<br>L_sdu:=Alarm_Ack-RES-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Res_SAP, L_sdu, Serv_class=Low,Transmit=Single) | AA-SRES |
| 84 | AA-WRES | MSAC1S_Alarm_Ack.rsp(-) ()<br>=><br>AA_State:=Upd<br>L_sdu:=Alarm_Ack-NRS-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Res_SAP, L_sdu, Serv_class=Low,Transmit=Single) | AA-SRES |
| 85 | AA-WRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 86 | AA-WRES | MSAC1S_Read.rsp(+)(Length, Data)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 87 | AA-WRES | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 88 | AA-WRES | MSAC1S_Write.rsp(+)(Length)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 89 | AA-WRES | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 90 | AA-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 91 | AA-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 92 | AA-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 93 | AA-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Read-REQ-PDU( ) && Res_SAP=Alarm_SAP<br>=><br>Service_Header:=Function_Num<br>MSAC1S_Read.ind(Req_Add=Loc_add, Slot_Number,Index,Length) | VS-WRES |
| 94 | AA-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Write-REQ-PDU( ) && Res_SAP=Alarm_SAP<br>=><br>Service_Header:=Function_Num<br>MSAC1S_Write.ind(Req_Add=Loc_add, Slot_Number,Index,Length,Data) | VS-WRES |
| 95 | AA-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT(Write-REQ-PDU( ) \|\| Read-REQ-PDU( )) \|\| Res_SAP=Server_SAP<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 96 | AA-WRES | MSAC1S_Start.req(Master_Add)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 97 | AA-WRES | MSAC1S_Stop.req<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | STOP-DEA51 |
| 98 | AA-WRES | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 99 | AA-WRES | MSAC1S_SInit_MS1.req<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 100 | AA-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Alarm_SAP,Serv_class,L_status)<br>/L-status=OK && AA_State=Upd && Res_SAP=Alarm_SAP<br>=><br>AA_State:=SRes | AA-SRES |
| 101 | AA-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L_status=LS/IV/LR<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 102 | AA-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Alarm_SAP,Serv_class,L_status)<br>/L-status=OK &&(AA_State<>Upd \|\| Res_SAP<>Alarm_SAP)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 103 | AA-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L-status=OK && AA_State=Upd && Res_SAP=Server_SAP<br>=><br>AA_State:=SRes | AA-SRES |
| 104 | AA-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L-status=OK && (AA_State<>Upd \|\| Res_SAP<>Server_SAP)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 105 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 106 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 107 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=SRes<br>=><br>AA_State:=Idle | OPEN |
| 108 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/(Update_status<>LO \|\| AA_State<>SRes) && Res_SAP=Alarm_SAP<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 109 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=SRes<br>=><br>AA_State:=Idle | OPEN |
| 110 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/(Update_status<>LO \|\| L_sdu.len<>0 \|\| AA_State<>SRes) && Res_SAP=Server<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 111 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Read-REQ-PDU( ) && RES_SAP=ALARM_SAP<br>=><br>Service_Header:=Function_Num<br>MSAC1S_Read.ind(Req_Add=Loc_add, Slot_Number,Index,Length) | VS-WRES |
| 112 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Write-REQ-PDU( ) && RES_SAP=ALARM_SAP<br>=><br>Service_Header:=Function_Num<br>MSAC1S_Write.ind(Req_Add=Loc_add, Slot_Number,Index,Length,Data) | VS-WRES |
| 113 | AA-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT(Write-REQ-PDU( ) \|\| Read-REQ-PDU( )) && RES_SAP=ALARM_SAP<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 114 | AA-SRES | MSAC1S_Start.req(Master_Add)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 115 | AA-SRES | MSAC1S_Stop.req<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | STOP-DEA51 |
| 116 | AA-SRES | MSAC1S_Read.rsp(+)(Length, Data)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 117 | AA-SRES | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 118 | AA-SRES | MSAC1S_Write.rsp(+)(Length)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 119 | AA-SRES | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 120 | AA-SRES | MSAC1S_Alarm_Ack.rsp (Alarm_Type, Slot_Number, Seq_Nr)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 121 | AA-SRES | MSAC1S_Alarm_Ack.rsp (Alarm_Type, Slot_Number, Seq_Nr)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 122 | AA-SRES | MSAC1S_SInit_MS1.req<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 123 | AA-SRES | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 124 | STOP-DEA51 | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status = OK<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Alarm_SAP) | STOP-DEA50 |
| 125 | STOP-DEA51 | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 126 | STOP-DEA50 | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Alarm_SAP,M_status)<br>/M_status = OK<br>=><br>MSAC1S_Stop.cnf | CLOSED |
| 127 | STOP-DEA50 | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Alarm_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 128 | ERR-DEA51 | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status = OK<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Alarm_SAP) | ERR-DEA50 |
| 129 | ERR-DEA51 | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 130 | ERR-DEA50 | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Alarm_SAP,M_status)<br>/M_status = OK | CLOSED |
| 131 | ERR-DEA50 | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Alarm_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC1S_Fault.ind | POWER-ON |

### 9.2.4 Functions

Table 56 contains the functions used by the MSAC1S, their arguments and their descriptions.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 56 – Functions used by the MSAC1S**

| Function name | Description |
|---|---|
| Read-REQ-PDU ( ) | This function will return TRUE if the corresponding L_sdu is a valid Read-REQ-PDU.<br>(L_sdu.len >= 4 && L_sdu[1] = 0x5E) |
| Write-REQ-PDU ( ) | This function will return TRUE if the corresponding L_sdu is a valid Write-REQ-PDU.<br>(L_sdu.len >= 4 && L_sdu[1] = 0x5F) |
| Alarm_Ack-REQ-PDU ( ) | This function will return TRUE if the corresponding L_sdu is a valid Alarm_Ack-REQ-PDU.<br>(L_sdu.len >= 4 && L_sdu[1] = 0x5C) |

## 9.3 SSCY1S

### 9.3.1 Primitive definitions

#### 9.3.1.1 Primitives exchanged between SSCY1S and FSPMS

Table 57 shows the service primitives including their associated parameters issued by the FSPMS and received by the SSCY1S.

**Table 57 – Primitives issued by FSPMS to SSCY1S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Get Publisher Data.req | FSPMS | Rem_Add | Refer to FAL Service Definition in IEC 61158-5-3 |
| Start Subscriber.req | FSPMS | Rem_Add | |
| Stop Subscriber.req | FSPMS | Rem_Add | |

Table 58 shows the service primitives including their associated parameters issued by the SSCY1S and received by the FSPMS.

**Table 58 – Primitives issued by SSCY1S to FSPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Get Publisher Data.cnf(+) | SSCY1S | Rem_Add,<br>Data,<br>New Flag | Refer to FAL Service Definition in IEC 61158-5-3 |
| Get Publisher Data.cnf(-) | SSCY1S | Rem_Add | |
| Start Subscriber.cnf(+) | SSCY1S | Rem_Add | |
| Start Subscriber.cnf(-) | SSCY1S | Rem_Add | |
| Stop Subscriber.cnf(+) | SSCY1S | Rem_Add | |
| Stop Subscriber.cnf(-) | SSCY1S | Rem_Add | |
| New Publisher Data.ind | SSCY1S | Rem_Add | |
| Publisher Active.ind | SSCY1S | Rem_Add,<br>Status | |

#### 9.3.1.2 Parameter of SSCY1S primitives

The parameters used with the primitives exchanged between the FSPMS and the SSCY1S are described in FAL Service Definition in IEC 61158-5-3.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 9.3.2 State machine description

After Power-On the SSCY1S waits for initiation by means of DMPMS DX Entered.ind with the Status TRUE. It indicates that the MSCY1S is opened for data exchange by the associated DP-master.

In the state W-START the service Start Subscriber called from the FSPMS is expected. When receiving this event the transition initializes all local variables and enters the RUN state. Furthermore, the local monitoring timer with the WD value (MSCY1S) is started.

When receiving a DMPMS DX Broadcast.ind with publisher data in the OPEN state the sample data according to the filter table are stored. Furthermore, the local flags New Data and New Stat will be set to TRUE and a New Publisher Data.ind will be issued to the FSPMS. The application retrieves the sample data with the Get Publisher Data request service primitive.

When the monitoring timer expires the status is checked and the timer will be started again. If the current status (New Stat) differs from status (Status) of the last interval the change is indicated to the application by means of the Publisher Active indication service primitive. Therefore, every change of status from active to passiv is reported to the application. The application may stop the subscribing procedure by sending a Stop Subscriber service request that sets the state machine back to W-START.

In any case, when a DMPMS DX Entered .ind with status FALSE (MSCY1S leaves the data exchange) is invoked the state machine is set back to POWER ON.

**Local variables of the SSCY1S**

**Status**
(Boolean)

This local variable is TRUE if the publisher was active during the last WD period. That means the data exchange mode was entered, the subsciber was started and at least one DXB had been received during the previous WD interval. Otherwise it is FALSE.

**New_Stat**
(Boolean)

This local variable is TRUE if the publisher is active. That means the Data Exchange is entered, the subsciber is started and at least one DXB has been received during the current WD interval. Otherwise it is FALSE

**New Data**
(Boolean)

This variable is TRUE if new data have arrived and are not retrieved by the Get Publisher Data service primitive. Otherwise it is FALSE.

**Sample Data**
(Octet-String)

The data buffer for publisher data.

### 9.3.3 SSCY1S state table

Table 59 contains the complete description of the SSCY1S state machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 59 – SSCY1S state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | SSCY1S_Start_Subscriber.req (Rem_Add)<br>=><br>SSCY1S_Start_Subscriber.cnf (-) (Rem_Add) | POWER-ON |
| 2 | POWER-ON | SSCY1S_Stop_Subscriber.req (Rem_Add)<br>=><br>SSCY1S_Stop_Subscriber.cnf (-) (Rem_Add) | POWER-ON |
| 3 | POWER-ON | SSCY1S_Get_Publisher_Data.req (Rem_Add)<br>=><br>SSCY1S_Get_Publisher_Data.cnf (-) (Rem_Add) | POWER-ON |
| 4 | POWER-ON | DMPMS_DX_Broadcast.ind (Rem_Add, Data)<br>=><br>ignore | POWER-ON |
| 5 | POWER-ON | DMPMS_DX_Entered.ind (Rem_Add, Status)<br>/Status<br>=> | W-START |
| 6 | W-START | SSCY1S_Start_Subscriber.req (Rem_Add)<br>=><br>SET_WD,<br>New_Stat := FALSE<br>New_Data := FALSE,<br>Sample_Data := 0<br>SSCY1S_Start_Subscriber.cnf (+) (Rem_Add) | RUN |
| 7 | W-START | SSCY1S_Stop_Subscriber.req (Rem_Add)<br>=><br>SSCY1S_Stop_Subscriber.cnf (-) (Rem_Add) | W-START |
| 8 | W-START | SSCY1S_Get_Publisher_Data.req (Rem_Add)<br>=><br>SSCY1S_Get_Publisher_Data.cnf (-) (Rem_Add) | W-START |
| 9 | W-START | DMPMS_DX_Broadcast.ind (Rem_Add, Data)<br>=><br>ignore | W-START |
| 10 | RUN | SSCY1S_Start_Subscriber.req (Rem_Add)<br>=><br>SSCY1S_Start_Subscriber.cnf (-) (Rem_Add) | RUN |
| 11 | RUN | SSCY1S_Stop_Subscriber.req (Rem_Add)<br>=><br>Stop Timer(WD)<br>SSCY1S_Stop_Subscriber.cnf (+) (Rem_Add) | W-START |
| 12 | RUN | SSCY1S_Get_Publisher_Data.req (Rem_Add)<br>=><br>Data := Sample_Data,<br>New_Flag := New_Data,<br>New_Data := FALSE<br>SSCY1S_Get_Publisher_Data.cnf (+) (Rem_Add, Data, New_Flag) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 13 | RUN | DMPMS_DX_Broadcast.ind (Rem_Add, Data)<br>=><br>New_Stat := TRUE,<br>New_Data := TRUE,<br>Sample_Data := Filter (Data)<br>SSCY1S_New_Publisher_Data.ind (Rem_Add) | RUN |
| 14 | RUN | WDTimer expired<br>/New_Stat<br>=><br>TRIG_WD,<br>New_Stat := FALSE<br>SSCY1S_Publisher_Active.ind (Rem_Add, Status) | RUN |
| 15 | RUN | WDTimer expired<br>/!New_Stat<br>=><br>TRIG_WD,<br>New_Stat := FALSE | RUN |
| 16 | any | DMPMS_DX_Entered.ind (Rem_Add, Status)<br>/!Status<br>=><br>Stop Timer(WD),<br>Publisher Address := 127,<br>Status := FALSE,<br>New_Stat := FALSE<br>SSCY1S_Publisher_Active.ind (Rem_Add, Status) | POWER-ON |

### 9.3.4 Functions

Table 60 contains the functions used by the SSCY1S, their arguments and their descriptions.

**Table 60 – Functions used by the SSCY1S**

| Function name | Description |
|---|---|
| Filter(Data) | This function will return sample data (Octete-String) extracted from Data according to the offset and length from the DXB link attributes. |
| SET_WD | if (WD Enabled = TRUE)<br>  Start Timer (WD)<br>else<br>  Stop Timer (WD) |
| TRIG_WD | if (WD Enabled = TRUE)<br>  Start Timer (WD) |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

## 9.4 MSRM2S

### 9.4.1 Primitive definitions

#### 9.4.1.1 Primitives exchanged between MSRM2S and FSPMS

Table 61 shows the service primitives including their associated parameters issued by the FSPMS and received by the MSRM2S.

**Table 61 – Primitives issued by FSPMS to MSRM2S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MS2.req | FSPMS | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMS | (none) | |

Table 62 shows the service primitives including their associated parameters issued by the MSRM2S and received by the FSPMS.

**Table 62 – Primitives issued by MSRM2S to FSPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MS2.cnf | MSRM2S | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | MSRM2S | (none) | |
| Fault.ind | MSRM2S | (none) | |

#### 9.4.1.2 Parameter of MSRM2S primitives

There are no parameters associated to the primitives.

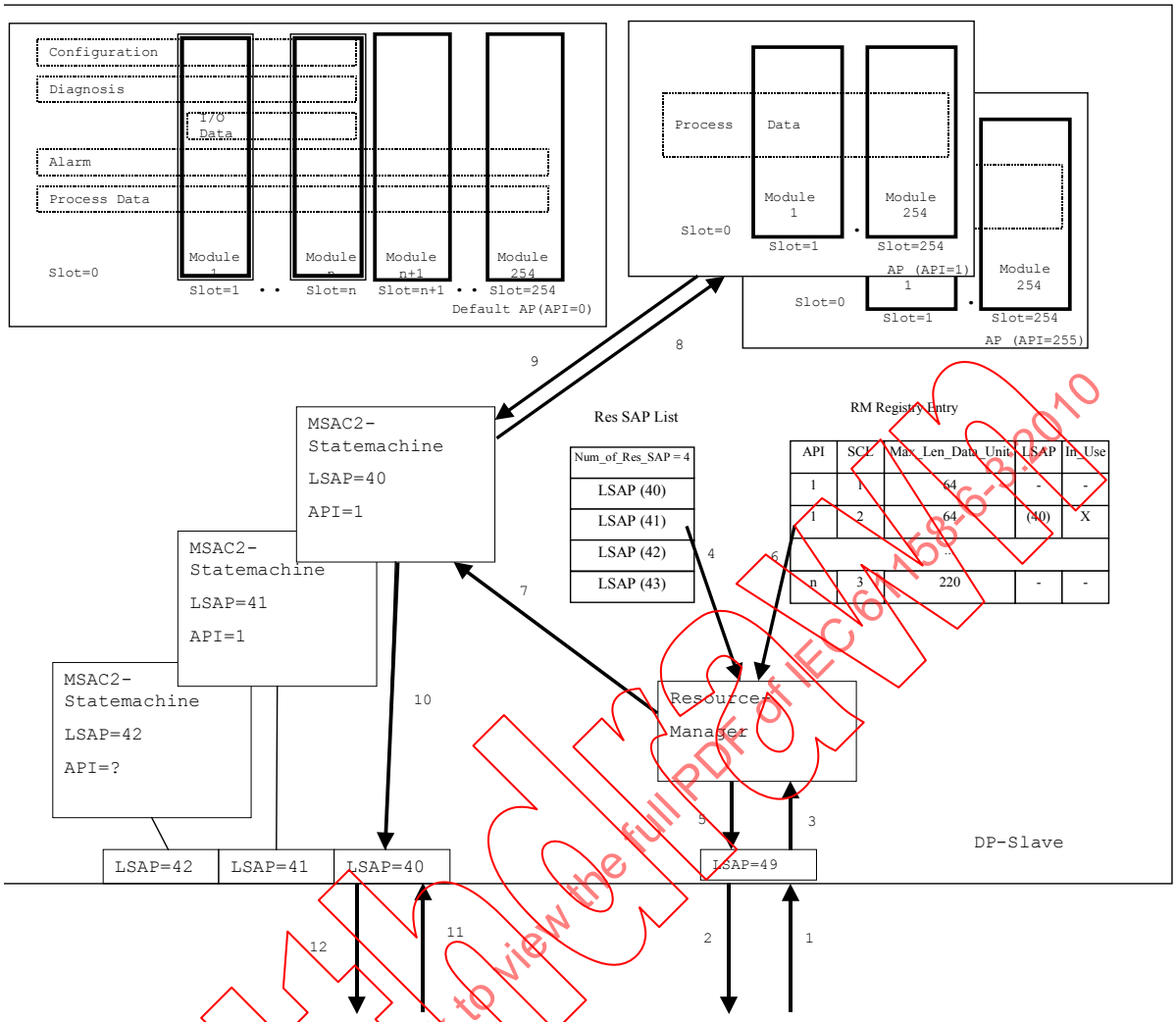
### 9.4.2 State machine description

For acyclic communication the DSAPs are allocated dynamically by the so called Resource Manager (RM) of the DP-slave.

The Resource Manager (RM) co-ordinates the communication resources for MS2 AR of the DP-slave. The main features of the RM are described in the following sequence, see Figure 27.

- The DP-master requests a communication connection by means of the service Initiate (DSAP 49). If multiple application processes are used in the DP-slave it is possible to address these application processes through an Application Process Identifier (API).
- The RM of the DP-slave provides an unused SAP (SAP 0 - 48). The possible SAPs are defined locally in the Res_SAP_List.
- The RM responds the Initiate.req immediately with the RM_REQ-PDU containing the corresponding unused SAP.
- Further steps following the Initiate.req (Initiate.res, Read, Write, Abort) are handled by the corresponding MSAC2S State Machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn




1 Initiate-REQ-PDU (DSAP=49)

2 RM-REQ-PDU (Server_SAP=next SAP not In_Use)

3 Initiate-REQ-PDU

4 Search of the next SAP not In_Use

5 Reply_Update.req(L_sdu=RM-REQ-PDU with next SAP not In_Use)

6 Check RM_Registry_Entry for API, In_Use=FALSE

7 MSAC2S_Initiate.req (Server_SAP=40)

8 MSAC2S_Initiate.ind (Server_SAP=40)

9 MSAC2S_Initiate.res (Server_SAP=40)

10 Reply_Update.req(L_sdu=Initiate-RES-PDU)

11 Polling with (L_sdu.len=0)

12 Initiate-RES-PDU

**Figure 27 – Example for connection establishment on MS2(server-side)**

The Resource Manager is responsible for the allocation of SAPs to incoming requests for a MS2 AR communication.

If a Res_SAP is available when an Initiate-REQ-PDU is received on the RM_SAP, the Slave responds with a RM-REQ-PDU.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

The only PDU which is accepted is an Initiate-REQ-PDU. The Resource Manager checks for an unused API/SCL in the RM_Registry and passes the Initiate-PDU with an Initiate to a MSAC2S State Machine.

If none of the entries in the RM_Registry fits to the requested API/SCL, the last SAP sent in the RM-REQ-PDU is temporarily used to send an Abort-REQ-PDU to the Master.

If no Res_SAP is available then the RM_SAP is deactivated.

If an MSAC2S State Machine indicates with anAbort.req that a connection has been closed, the corresponding entry in the RM_Registry is marked as unused and the corresponding DLSAP is released and can be used for further requests.

**Local variables of the MSRM2S**

**List of RM entries**

Figure 28 shows the structure of the RM entries as they are listed in the RM_Registry.

| **static part** | | | **dynamic part** | |
|---|---|---|---|---|
| API | SCL | Max_Len_Data_Unit | SAP | In_Use |

**Figure 28 – Structure of RM entries in the RM_Registry**

**List of SSAPs**

**SSAP_Entry**
(Unsigned8)

0..48

**Stored_Req_Add,**
**Stored_Send_Timeout,**
**Stored_Features_Supported,**
**Stored_Profile_Features_Supported,**
**Stored_Profile_Ident_Number,**
**Stored_Add_Addr_Param**
Storage of values of the last Initiate function (see part DP-Services).

**RM_SAP**
(constant)

Set to 49

**Last_SAP**
(Unsigned8)

0..48

SAP for previous established connection.

**Server_SAP**
(Unsigned8)

0..48

SAP which will be allocated to the next connection.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Macros**

**INITIATE-REQ-PDU-LEN**
(

48 {64 for linking device}

)

**RMREQ-PDU-LEN**
(

4

)

### 9.4.3 MSRM2S state table

Table 63 contains the complete description of the MSRM2S state machine.

**Table 63 – MSRM2S state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | MSRM2S_SInit.req<br>=>RM_SAP:=49<br>Res_SAP_entry_Index:=1<br>Server_SAP:=Get_from_List_of_SSAPs()<br>DMPMS_RSAP_ACTIVATE.req(SSAP=RM_SAP, Access=All,<br>L_sdu_length_list=(<INITIATE-REQ-PDU-LEN>,0;<RMREQ-PDU-LEN>,0)<br>Indication_Mode=Data) | WAIT-ACT |
| 2 | WAIT-ACT | DMPMS_RSAP_ACTIVATE.cnf(SSAP,M_status)<br>/M_status=OK<br>=>RM_SAP:=49<br>Res_SAP_entry_Index:=1<br>Server_SAP:=Get_from_List_of_SSAPs()<br>DMPMS_REPLY_UPDATE.req(SSAP=RM_SAP, L_sdu=RM-REQ-PDU,<br>Serv_class=Low, Transmit=Single) | INIT-WUPD |
| 3 | WAIT-ACT | DMPMS_RSAP_ACTIVATE.cnf(SSAP,M_status)<br>/M_status=NO/IV<br>=>MSRM2S_Fault.ind | POWER-ON |
| 4 | INIT-WUPD | DMPMS_REPLY_UPDATE.cnf (SSAP=RM_SAP, Serv_class=Low, L_status)<br>/L_status=OK<br>=>MSRM2S_SInit.cnf | OPEN |
| 5 | INIT-WUPD | DMPMS_REPLY_UPDATE.cnf (SSAP=RM_SAP, Serv_class=Low, L_status)<br>/L_status= LS/IV/LR<br>=>MSRM2S_Fault.ind | POWER-ON |
| 6 | OPEN | DMPMS_DATA_REPLY.ind(SSAP, DSAP=RM_SAP, Loc_add, Rem_add,L_sdu,<br>Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu=Initiate-REQ-PDU && List_of_SSAPs<>empty<br>=>Last_SAP := Server_SAP<br>Stored_Req_Add:=Loc_add,<br>Stored_Send_Timeout:= Send_Timeout<br>Stored_Features_Supported:=Features_Supported<br>Stored_Profile_Features_Supported:= Profile_Features_Supported<br>Stored_Profile_Ident_Number:=Profile_Ident_Number<br>Stored_Add_Addr_Param:= Add_Addr_Param<br>Server_SAP:=Get_from_List_of_SSAPs()<br>DMPMS_REPLY_UPDATE.req(SSAP=RM_SAP, L_sdu=RM-REQ-PDU,<br>Serv_class=Low, Transmit=Single) | OW-UPDATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 7 | OPEN | DMPMS_DATA_REPLY.ind(SSAP, DSAP=RM_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu=Initiate-REQ-PDU && List_of_SSAPs=empty<br>=>Stored_Req_Add:=Loc_add,<br>Stored_Send_Timeout:= Send_Timeout<br>Stored_Features_Supported:=Features_Supported<br>Stored_Profile_Features_Supported:= Profile_Features_Supported<br>Stored_Profile_Ident_Number:=Profile_Ident_Number<br>Stored_Add_Addr_Param:= Add_Addr_Param<br>DMPMS_SAP_DEACTIVATE.req( SSAP=RM_SAP) | OW-DEACT |
| 8 | OPEN | DMPMS_DATA_REPLY.ind(SSAP, DSAP=RM_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && (L_sdu<>Initiate-REQ-PDU)<br>=>ignore Lsdu<br>DMPMS_REPLY_UPDATE.req(SSAP=RM_SAP, L_sdu=RM-REQ-PDU, Serv_class=Low, Transmit=Single) | INIT-WUPD |
| 9 | OPEN | DMPMS_DATA_REPLY.ind(SSAP, DSAP=RM_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=NO<br>=>ignore Lsdu | OPEN |
| 10 | OPEN | MSRM2S_Reset.req<br>=>for (all CRL entries with AR-Type=MS2) MSAC2S_Reset.req(Res_SAP=SSAP ) endfor | RESET-AC2 |
| 11 | OPEN | MSAC2S_Closed.ind(Res_SAP)<br>=>Index := Search_in_List_of_RM_Entries(SAP:=Res_SAP, In_Use:=True) | AW-SEARCH |
| 12 | AW-SEARCH | /Index <> 0<br>=>List_of_RM_Entries.In_Use[Index]:=False<br>Put_to_List_of_SSAPs(Stored_Res_SAP) | OPEN |
| 13 | AW-SEARCH | /Index = 0<br>=>MSRM2S_Fault.ind | POWER-ON |
| 14 | OW-UPDATE | DMPMS_REPLY_UPDATE.cnf (SSAP=RM_SAP, Serv_class=Low, L_status)<br>/L_status=OK<br>=>Index := Search_in_List_of_RM_Entries(API:=Add_Addr_Param.D_Addr.API, SCL:= Add_Addr_Param.D_Addr.SCL, In_Use:=False) | OW-SEARCH |
| 15 | OW-UPDATE | DMPMS_REPLY_UPDATE.cnf (SSAP=RM_SAP, Serv_class=Low, L_status)<br>/L_status= LS/IV/LR<br>=>MSRM2S_Fault.ind | POWER-ON |
| 16 | OW-SEARCH | /Index <> 0<br>=>List_of_RM_Entries.In_Use[Index]:=True<br>List_of_RM_Entries.SAP[Index]:=Last_SAP<br>Max_Len_Data_Unit:=List_of_RM_Entries.Max_Len_Data_Unit[Index]<br>Send_Timeout:=Max(Stored_Send_Timeout, Stored_Min_Send_Timeout)<br>MSAC2S_Initiate.req(Res_SAP=Last_SAP, Req_Add=Stored_Req_Add,<br>Max_Len_Data_Unit,<br>Send_Timeout, Features_Supported= Stored_Features_Supported,<br>Profile_Features_Supported= Stored_Profile_Features_Supported,<br>Profile_Ident_Number= Stored_Profile_Ident_Number,<br>Add_Addr_Param= Stored_Add_Addr_Param) | OPEN |
| 17 | OW-SEARCH | /Index= 0<br>=>Send_Timeout:=Max(Stored_Send_Timeout, Stored_Min_Send_Timeout)<br>MSAC2S_Abort.req(Res_SAP=Last_SAP, Req_Add, Subnet=NO,<br>Instance=MSAC2, Reason_Code=ABT_IA, Send_Timeout) | OPEN |
| 18 | OW-DEACT | DMPMS_SAP_DEACTIVATE.cnf(SSAP,M_status)<br>/M_status=OK<br>=>Index := Search_in_List_of_RM_Entries(API:=Add_Addr_Param.D_Addr.API, SCL:= Add_Addr_Param.D_Addr.SCL, In_Use:=False) | CW-SEARCH |
| 19 | OW-DEACT | DMPMS_SAP_DEACTIVATE.cnf(SSAP,M_status)<br>/M_status=NO/IV<br>=>MSRM2S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 20 | CW-SEARCH | /Index <> 0<br>=>List_of_RM_Entries.In_Use[Index]:=True<br>List_of_RM_Entries.SAP[Index]:=Last_SAP<br>Max_Len_Data_Unit:=List_of_RM_Entries.Max_Len_Data_Unit[Index]<br>Send_Timeout:=Max(Stored_Send_Timeout, Stored_Min_Send_Timeout)<br>MSAC2S_Initiate.req(Res_SAP=Last_SAP, Req_Add=Stored_Req_Add,<br>Max_Len_Data_Unit,<br>Send_Timeout, Features_Supported= Stored_Features_Supported,<br>Profile_Features_Supported= Stored_Profile_Features_Supported,<br>Profile_Ident_Number= Stored_Profile_Ident_Number,<br>Add_Addr_Param= Stored_Add_Addr_Param) | CLOSED |
| 21 | CW-SEARCH | /Index = 0<br>=>Send_Timeout:=Max(Stored_Send_Timeout, Stored_Min_Send_Timeout)<br>MSAC2S_Abort.req(Res_SAP=Last_SAP, Req_Add, Subnet=NO,<br>Instance=MSAC2, Reason_Code=ABT_IA, Send_Timeout) | CLOSED |
| 22 | CLOSED | MSRM2S_Reset.req<br>=>for (all CRL entries with AR-Type=MS2) MSAC2S_Reset.req(Res_SAP=SSAP<br>) endfor | RESET-AC2 |
| 23 | CLOSED | MSAC2S_Closed.ind(Res_SAP)<br>=>Stored_Res_SAP:=Res_SAP<br>Index := Search_in_List_of_RM_Entries(SAP:=Res_SAP, In_Use:=True) | REOW-SEARCH |
| 24 | REOW-SEARCH | /Index <> 0<br>=>List_of_RM_Entries.In_Use[Index]:=False<br>Put_to_List_of_SSAPs(Stored_Res_SAP) | WAIT-ACT |
| 25 | REOW-SEARCH | /Index = 0<br>=>MSRM2S_Fault.ind | POWER-ON |
| 26 | RESET-AC2 | for (all CRL entries with AR-Type=MS2) MSAC2S_Reset.cnf(Res_SAP=SSAP )<br>endfor<br>/M_status=OK<br>=>MSRM2S_Reset.cnf | POWER-ON |
| 27 | ANY-STATE | MSAC2S_Fault.ind(Res_SAP)<br>=>MSRM2S_Fault.ind | POWER-ON |
| 28 | ANY-STATE | unexpected DL reaction<br>=>MSRM2S_Fault.ind | POWER-ON |

## 9.5 MSAC2S

### 9.5.1 Primitive definitions

#### 9.5.1.1 Primitives exchanged between MSAC2S and FSPMS

Table 64 shows the service primitives including their associated parameters issued by the FSPMS and received by the MSAC2S.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 64 – Primitives issued by FSPMS to MSAC2S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Initiate.rsp(+) | FSPMS | Res SAP,<br>Max Len Data Unit,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | Refer to FAL Service Definition in IEC 61158-5-3 |
| Initiate.rsp(-) | FSPMS | Res SAP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Read.rsp(+) | FSPMS | Res SAP,<br>Length,<br>Data | |
| Read.rsp(-) | FSPMS | Res SAP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.rsp(+) | FSPMS | Res SAP,<br>Length | |
| Write.rsp(-) | FSPMS | Res SAP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Data Transport.rsp(+) | FSPMS | Res SAP,<br>Length,<br>Data | |
| Data Transport.rsp(-) | FSPMS | Res SAP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Abort.req | FSPMS | Res SAP,<br>Subnet,<br>Instance,<br>Reason Code | |

Table 65 shows the service primitives including their associated parameters issued by the MSAC2S and received by the FSPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 65 – Primitives issued by MSAC2S to FSPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Initiate.ind | MSAC2S | Res SAP,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | Refer to FAL Service Definition in IEC 61158-5-3 |
| Abort.ind | MSAC2S | Res SAP,<br>Locally Generated,<br>Subnet,<br>Instance,<br>Reason Code,<br>Additional Detail | |
| Read.ind | MSAC2S | Res SAP,<br>Slot Number,<br>Index,<br>Length | |
| Write.ind | MSAC2S | Res SAP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Data Transport.ind | MSAC2S | Res SAP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |

#### 9.5.1.2 Primitives exchanged between MSAC2S and MSRM2S

Table 66 shows the service primitives including their associated parameters issued by MSAC2S and received by the MSRM2S.

**Table 66 – Primitives issued by MSRM2S to MSAC2S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Reset.req | MSRM2S | Res SAP | Refer to FAL Service Definition in IEC 61158-5-3 |
| Initiate.req | MSRM2S | Res SAP,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| RM2_Abort.req | MSRM2S | Res SAP,<br>Subnet,<br>Instance,<br>Reason Code | |

Table 67 shows the service primitives including their associated parameters issued by MSAC2S and received by the MSRM2S.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 67 – Primitives issued by MSAC2S to MSRM2S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Reset.cnf | MSAC2S | Res SAP | Refer to FAL Service Definition in IEC 61158-5-3 |
| Fault.ind | MSAC2S | Res SAP | |
| Closed.ind | MSAC2S | Res SAP | Signals the Termination of a MS2 Communication |

#### 9.5.1.3 Parameters of MSAC2S primitives

The parameters used with the primitives exchanged between the MSAC2S and FSPMS,MSRM2S are described in FAL Service Definition in IEC 61158-5-3. Additional parameters are described in Table 68.

**Table 68 – Parameter used with primitives exchanged with MSAC2S**

| Parameter name | Description |
|---|---|
| Res SAP | This parameter conveys the SSAP for addressing of the various MS2 AR. |

### 9.5.2 State machine description

In the state CLOSED the Initiate service is expected from the Resource Manager. Then the corresponding DLSAP is activated and the User gets the Initiate.ind. The User shall generate a MSAC2S_Initiate.rsp and the connection is established after the Master has fetched the response.

In the OPEN-state valid services (see macro <VALID_SERVICE>) are allowed. Only one request from the DP-master is processed simultaneously, otherwise the connection is aborted.

Both the DP-master and the DP-slave can close the connection using the Abort. The remote User gets an Abort.ind.

The connection monitoring is based on the U-Timer, F-Timer and I-Timer (see 6.9).

**Local variables of the MSAC2S**

**Server_SAP**
(Unsigned8)

The used SAP for the MS2 connection.

**U-Timer**
(Unsigned16)

The U-Timer controls the reaction of the User after delivering an indication. If the U-Timer expires an Idle-PDU is created and transferred to the local DLL. The U-Timer is stopped when the User provides the response.

**F-Timer**
(Unsigned16)

The F-Timer controls the Master fetching the previously provided PDU. If the F-Timer expires an Abort is generated, transferred to the local DLL and the F-Timer starts again. The F-Timer is stopped as soon as the Master fetches the PDU.

**I-Timer**
(Unsigned16)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

The I-Timer controls the next activity of the client after the Master has fetched the previous PDU. When the I-Timer expires an Abort is generated, transferred to the local DLL and the F-Timer starts. The I-Timer is stopped when the Master has sent the next request.

NOTE Only one of the timers described above is running at the same time.

**Service_Header**
(Unsigned8)

The actual service header to check whether the User generates the correct response.

**Return-State**
INITIATE, VALID-SERVICE

This variable is used to store the state where to return after an idle cycle has been processed.

**Go_Abort**
(Boolean)

Local flag that indicates an abort reaction after finishing the present service response.

**Service_Buffer**
(Octet-String[244])

Local buffer that stores one complete Response-PDU (Initiate.rsp or <VALID_SERVICE>.rsp) during an outstanding Idle-RES-PDU.

**Stored_Slot_Number**
Actual value of the slot number

**Stored_Index**
Stored values of the parameters of Read/Write-services

**Stored_Instance**
Actual value of the instance (e.g. MSAC2)

**Stored_Reason**
Stored values of the parameters of MSAC2S_Abort-service

**INITIATE-REQ-PDU-LEN**
(local constant)

PDU-size (the 4 Octet Header is excluded) set to 48 (64 respectively for links)

**ABORT_PDU_LEN**
(local constant)

Length of an Abort-REQ-PDU = 4

**Macros**

**Read-REQ-PDU**
This function will return True if the corresponding L_sdu is a valid Read-REQ-PDU

**Write-REQ-PDU**
This function will return True if the corresponding L_sdu is a valid Write-REQ-PDU

**Data_Transport-REQ-PDU**
This function will return True if the corresponding L_sdu is a valid Data_Transport-REQ-PDU

**Abort-REQ-PDU**
This function will return True if the corresponding L_sdu is a valid Abort-REQ-PDU

**Idle-REQ-PDU**
This function will return True if the corresponding L_sdu is a valid Idle-REQ-PDU

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**STORE_ABORT_PARAMETER**
(

Stored-Instance=Instance

Stored-Reason_Code=Reason_Code

)

**<VALID-SERVICE>**
<

Service_Header=0x5E: Read

Service_Header=0x5F: Write

Service_Header=0x51: Data_Transport

>

**VALID-SERVICE-REQ-PDU**
(

(Read-REQ-PDU || Write-REQ-PDU || OR Data_Transport-REQ-PDU)

&& L_sdu.len >= 4

&& (L_sdu[1] = 0x5e || (L_sdu.len >= Length+4

&& Max_DLSDU_length_ind_low-4 >= Length)

)

### 9.5.3 MSAC2S state table

Table 69 contains the complete description of the MSAC2S state machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 69 – MSAC2S state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | =><br>Server_SAP:=Res_SAP | CLOSED |
| 2 | CLOSED | MSAC2S_Initiate.req(Res_SAP, Req_Add, Max_Len_Data_Unit, Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>Service_Header:=0x57<br>F-Timer:=Send_Timeout<br>U-Timer:=Send_Timeout<br>I-Timer:=2*Send_Timeout<br>Stored_Max_Len_Data_Unit:= Max(Max_Len_Data_Unit, INITIATE-REQ-PDU-LEN)<br>clear Service_Buffer<br>Go_Abort:=FALSE<br>MSAC2_Initiate.ind(Req_Add, Res_SAP, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>DMPMS_RSAP_ACTIVATE.req(SSAP=Server_SAP, Access=Req_Add, L_sdu_length_list=(Stored_Max_Len_Data_Unit+4,0, Stored_Max_Len_Data_Unit+4,0) , Indication_Mode=Data) | SET-ACC-POS |
| 3 | CLOSED | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>=><br>ignore | CLOSED |
| 4 | CLOSED | MSAC2S_RM2_Abort.req(Res_SAP, Req_Add, Subnet, Instance, Reason_Code, Send_Timeout)<br>=><br>F-Timer:=Send_Timeout<br>STORE_ABORT_PARAMETER<br>DMPMS_RSAP_ACTIVATE.req(SSAP=Server_SAP, Access=Req_add, L_sdu_length_list=(ABORT_PDU_LEN,0, ABORT_PDU_LEN,0) Indication_Mode=Data) | SET-ACC-FE |
| 5 | CLOSED | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>=><br>Reason_Code:=ABT_SE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | CLOSED |
| 6 | CLOSED | MSAC2S_Initiate.rsp(+/-)( Res_SAP)<br>=><br>Reason_Code:=ABT_SE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | CLOSED |
| 7 | CLOSED | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>ignore | CLOSED |
| 8 | SET-ACC-POS | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status = OK<br>Start U-Timer | INITI-WRES |
| 9 | SET-ACC-POS | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC2S_Fault.ind | POWER-ON |
| 10 | SET-ACC-FE | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status = OK<br>=><br>L_sdu:=Abort-REQ-PDU(with stored parameter)<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 11 | SET-ACC-FE | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC2S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 12 | INITI-WRES | MSAC2S_Initiate.rsp(+)(Res_SAP)<br>=><br>L_sdu:=Initiate-RES-PDU(With Stored_MAx_Len_Data_Unit)<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | INITI-WUPD |
| 13 | INITI-WRES | MSAC2S_Initiate.rsp(-)(Res_SAP)<br>=><br>L_sdu:=Initiate-NRS-PDU<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 14 | INITI-WRES | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>=><br>Reason_Code:=ABT_SE<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 15 | INITI-WRES | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>L_sdu:=Abort-REQ-PDU<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 16 | INITI-WRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Abort-REQ-PDU<br>=><br>Stop U-Timer<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=FALSE, (Subnet, Instance, Reason_Code) from Abort-REQ-PDU) | WAIT-DEACT |
| 17 | INITI-WRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/NOT(Abort-REQ-PDU)<br>=><br>Reason_Code:=ABT_FE<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 18 | INITI-WRES | U-Timer expired<br>=><br>L_sdu:=Idle-REQ-PDU<br>Return-State:=INITIATE<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | IDLE-WUPD |
| 19 | INITI-WUPD | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP, Serv_class=Low, L_status)<br>/L_status=OK<br>=><br>ignore | INITI-SRES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 20 | INITI-WUPD | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP, Serv_class=Low, L_status)<br>/L_status= LS/IV/LR<br>=><br>Stop F-Timer<br>Stop I-Timer<br>MSAC2S_Fault.ind | POWER-ON |
| 21 | INITI-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len=0 && Go_Abort = FALSE<br>=><br>Stop F-Timer<br>Start I-Timer | OPEN |
| 22 | INITI-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len=0 && Go_Abort = TRUE<br>=><br>L_sdu:=Abort-REQ-PDU(With Stored Parameter)<br>Stop F-Timer<br>Start I-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 23 | INITI-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO/NO && L_sdu.len<>0 && Abort-REQ-PDU<br>=><br>Stop F-Timer<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=FALSE, (Subnet, Instance, Reason_Code) from Abort-REQ-PDU) | WAIT-DEACT |
| 24 | INITI-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len<>0 && NOT(Abort-REQ-PDU)<br>=><br>Reason_Code:=ABT_FE<br>Stop F-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 25 | INITI-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=NO && L_sdu.len<>0 && NOT(Abort-REQ-PDU)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_FE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | INITI-SRES |
| 26 | INITI-SRES | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_SE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | INITI-SRES |
| 27 | INITI-SRES | MSAC2S_Initiate.rsp(+/-)( Res_SAP)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_SE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | INITI-SRES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 28 | INITI-SRES | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER | INITI-SRES |
| 29 | INITI-SRES | F-Timer expired<br>=><br>Reason_Code:=ABT_TO<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 30 | IDLE-WUPD | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP, Serv_class=Low, L_status)<br>/Status=OK | IDLE-SREQ |
| 31 | IDLE-WUPD | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP, Serv_class=Low, L_status)<br>/L_status= LS/IV/LR<br>=><br>Stop F-Timer<br>Stop I-Timer<br>MSAC2S_Fault.ind | POWER-ON |
| 32 | IDLE-SREQ | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len=0 && Go_Abort = FALSE<br>=><br>Stop F-Timer<br>Start I-Timer | W-IDLE-CON |
| 33 | IDLE-SREQ | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len=0 && Go_Abort = TRUE<br>=><br>L_sdu:=Abort-REQ-PDU(With Stored Parameter)<br>Stop F-Timer<br>Start I-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 34 | IDLE-SREQ | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO/NO && L_sdu.len<>0 && Abort-REQ-PDU<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2_Abort.ind(Res_SAP=Server_SAP, Locally-Generated=FALSE, (Subnet, Instance, Reason_Code) from Abort-REQ-PDU)<br>Stop F-Timer | WAIT-DEACT |
| 35 | IDLE-SREQ | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len<>0 && NOT(Abort-REQ-PDU)<br>=><br>Reason_Code:=ABT_FE<br>Stop F-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 36 | IDLE-SREQ | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=NO && L_sdu.len<>0 && NOT(Abort-REQ-PDU)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_FE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | IDLE-SREQ |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 37 | IDLE-SREQ | MSAC2S_XXX.rsp(+)(Res_SAP)<br>/Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length && Service_Buffer empty<br>=><br>Service_Buffer:= <VALID-SERVICE>-RES-PDU(with Stored_Slot_Number,Stored_Index) | IDLE-SREQ |
| 38 | IDLE-SREQ | MSAC2S_XXX.rsp(-)(Res_SAP)<br>/Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length && Service_Buffer empty<br>=><br>Service_Buffer:= <VALID-SERVICE>-NRS-PDU(with Stored_Slot_Number,Stored_Index) | IDLE-SREQ |
| 39 | IDLE-SREQ | MSAC2S_Initiate.rsp(+)(Res_SAP)<br>/Service-Response fits stored Service_Header && Service_Buffer empty<br>=><br>Service_Buffer:= Initiate-RES-PDU | IDLE-SREQ |
| 40 | IDLE-SREQ | MSAC2S_Initiate.rsp(-)(Res_SAP)<br>/Service-Response fits stored Service_Header && Service_Buffer empty<br>=><br>Service_Buffer:= Initiate-NRS-PDU | IDLE-SREQ |
| 41 | IDLE-SREQ | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>/(NOT(Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length ) \|\| NOT(Service_Buffer empty))<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_RE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | IDLE-SREQ |
| 42 | IDLE-SREQ | MSAC2S_Initiate.rsp(+/-)( Res_SAP)<br>/(NOT(Service-Response fits stored Service_Header ) \|\| NOT(Service_Buffer empty))<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_RE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | IDLE-SREQ |
| 43 | IDLE-SREQ | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER | IDLE-SREQ |
| 44 | IDLE-SREQ | F-Timer expired<br>=><br>Reason_Code:=ABT_TO<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 45 | W-IDLE-CON | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Idle-RES-PDU && Return-State=INITIATE && Go_Abort=FALSE && (Service_Buffer empty)<br>=><br>Stop I-Timer<br>Start U-Timer | INITI-WRES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 46 | W-IDLE-CON | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Idle-RES-PDU && Return-State=INITIATE && Go_Abort=FALSE && NOT(Service_Buffer empty) && Service_Buffer=Initiate-NRS-PDU<br>=><br>L_sdu:=Stored L_sdu from Service_Buffer<br>clear Service_Buffer<br>Stop I-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 47 | W-IDLE-CON | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Idle-RES-PDU && Return-State=INITIATE && Go_Abort=FALSE && NOT(Service_Buffer empty) && Service_Buffer=Initiate-RES-PDU<br>=><br>L_sdu:=Stored L_sdu from Service_Buffer<br>clear Service_Buffer<br>Stop I-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | INITI-WUPD |
| 48 | W-IDLE-CON | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Idle-RES-PDU && Return-State=VALID-SERVICE && Go_Abort=FALSE && (Service_Buffer empty)<br>=><br>Stop I-Timer<br>Start U-Timer | VS-WRES |
| 49 | W-IDLE-CON | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Idle-RES-PDU && Return-State=VALID-SERVICE && Go_Abort=FALSE && NOT(Service_Buffer empty)<br>=><br>L_sdu:=Stored L_sdu from Service_Buffer<br>clear Service_Buffer<br>Stop I-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | VS-WUPD |
| 50 | W-IDLE-CON | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Idle-RES-PDU && Go_Abort=TRUE<br>=><br>L_sdu:=Abort-REQ-PDU( with stored Parameter)<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 51 | W-IDLE-CON | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Abort-REQ-PDU<br>=><br>Stop I-Timer<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=FALSE, (Subnet, Instance, Reason_Code) from Abort-REQ-PDU) | WAIT-DEACT |
| 52 | W-IDLE-CON | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/NOT(Abort-REQ-PDU \|\| Idle-RES-PDU)<br>=><br>Reason_Code:=ABT_FE<br>Stop I-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 53 | W-IDLE-CON | MSAC2S_XXX.rsp(+)(Res_SAP)<br>/Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length && Service_Buffer empty<br>=><br>Service_Buffer:= <VALID-SERVICE>-RES-PDU(with Stored_Slot_Number,Stored_Index) | W-IDLE-CON |
| 54 | W-IDLE-CON | MSAC2S_XXX.rsp(-)(Res_SAP)<br>/Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length && Service_Buffer empty<br>=><br>Service_Buffer:= <VALID-SERVICE>-NRS-PDU (with Stored_Slot_Number,Stored_Index) | W-IDLE-CON |
| 55 | W-IDLE-CON | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>/(NOT(Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length) \|\| NOT(Service_Buffer empty))<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_RE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | W-IDLE-CON |
| 56 | W-IDLE-CON | MSAC2S_Initiate.rsp(+/-)( Res_SAP)<br>/(NOT(Service-Response fits stored Service_Header ) \|\| NOT(Service_Buffer empty))<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_RE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | W-IDLE-CON |
| 57 | W-IDLE-CON | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER | W-IDLE-CON |
| 58 | W-IDLE-CON | I-Timer expired<br>=><br>Reason_Code:=ABT_TO<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 59 | SABORT-WUPD | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP, Serv_class=Low, L_status)<br>/Status=OK | SABORT-SRES |
| 60 | SABORT-WUPD | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP, Serv_class=Low, L_status)<br>/L_status= LS/IV/LR<br>=><br>Stop F-Timer<br>Stop I-Timer<br>MSAC2S_Fault.ind | POWER-ON |
| 61 | SABORT-SRES | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>ignore | SABORT-SRES |
| 62 | SABORT-SRES | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>=><br>Reason_Code:=ABT_SE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-SRES |
| 63 | SABORT-SRES | MSAC2S_Initiate.rsp(+/-)( Res_SAP)<br>=><br>Reason_Code:=ABT_SE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-SRES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 64 | SABORT-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len=0<br>=><br>Stop I-Timer<br>Stop F-Timer<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | WAIT-DEACT |
| 65 | SABORT-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=NO && L_sdu.len<>0<br>=><br>ignore | SABORT-SRES |
| 66 | SABORT-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len<>0<br>=><br>ignore L_sdu<br>Stop I-Timer<br>Stop F-Timer<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | WAIT-DEACT |
| 67 | SABORT-SRES | I-Timer expired<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code=ABT_TO) | WAIT-DEACT |
| 68 | SABORT-SRES | F-Timer expired<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code=ABT_TO) | WAIT-DEACT |
| 69 | WAIT-DEACT | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status = OK<br>=><br>MSAC2S_Closed.ind(Res_SAP) | CLOSED |
| 70 | WAIT-DEACT | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC2S_Fault.ind | POWER-ON |
| 71 | OPEN | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/VALID-SERVICE-REQ-PDU<br>=><br>Service_Header:=Function_Num<br>Stored_Slot_Number:=Slot_Number<br>Stored_Index:=Index<br>Stop I-Timer<br>Start U-Timer<br>MSAC2S_<VALID-SERVICE>.ind(Res_SAP=Server_SAP) | VS-WRES |
| 72 | OPEN | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Idle-REQ-PDU {IDLE-PDU}<br>=><br>L_sdu:=Idle-RES-PDU<br>Stop I-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | VS-WUPD |
| 73 | OPEN | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Abort-REQ-PDU<br>=><br>Stop I-Timer<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=FALSE, (Subnet, Instance, Reason_Code) from Abort-REQ-PDU) | WAIT-DEACT |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 74 | OPEN | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/NOT(VALID-SERVICE-REQ-PDU \|\| (Abort-REQ-PDU) \|\| (Idle-REQ-PDU))<br>=><br>Reason_Code:=ABT_FE<br>Stop I-Timer<br>Start I-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 75 | OPEN | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>L_sdu:=Abort-REQ-PDU<br>Stop I-Timer<br>Start I-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 76 | OPEN | MSAC2S_Initiate.rsp(+/-)( Res_SAP)<br>=><br>Reason_Code:=ABT_SE<br>Stop I-Timer<br>Start I-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 77 | OPEN | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>=><br>L_sdu:=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code=ABT_TO)<br>Stop I-Timer<br>Start I-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 78 | OPEN | I-Timer expired<br>=><br>Reason_Code:=ABT_TO<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 79 | VS-WRES | MSAC2S_XXX.rsp(+)(Res_SAP)<br>/Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length<br>=><br>L_sdu:=<VALID-SERVICE>-RES-PDU(with Stored_Slot_Number,Stored_Index)<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | VS-WUPD |
| 80 | VS-WRES | MSAC2S_XXX.rsp(-)(Res_SAP)<br>/Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length<br>=><br>L_sdu:=<VALID-SERVICE>-NRS-PDU<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | VS-WUPD |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 81 | VS-WRES | MSAC2S_Initiate.rsp(+/-)( Res_SAP)<br>=><br>Reason_Code:=ABT_SE<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 82 | VS-WRES | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>/NOT(Service-Response fits stored Service_Header && Stored_Max_Len_Data_Unit >= Length)<br>=><br>Reason_Code:=ABT_RE<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 83 | VS-WRES | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>L_sdu:=Abort-REQ-PDU<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 84 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Abort-REQ-PDU<br>=><br>Stop U-Timer<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=FALSE, (Subnet, Instance, Reason_Code) from Abort-REQ-PDU) | WAIT-DEACT |
| 85 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/NOT(Abort-REQ-PDU)<br>=><br>Reason_Code:=ABT_FE<br>Stop U-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 86 | VS-WRES | U-Timer expired<br>=><br>L_sdu:=Idle-REQ-PDU<br>Return-State:=VALID-SERVICE<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | IDLE-WUPD |
| 87 | VS-WUPD | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP, Serv_class=Low, L_status)<br>/Status=OK | VS-SRES |
| 88 | VS-WUPD | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP, Serv_class=Low, L_status)<br>/L_status= LS/IV/LR<br>=><br>Stop F-Timer<br>Stop I-Timer<br>MSAC2S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 89 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len=0 && Go_Abort = FALSE<br>Stop F-Timer<br>Start I-Timer | OPEN |
| 90 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len=0 && Go_Abort = TRUE<br>=><br>L_sdu:=Abort-REQ-PDU with stored Parameter<br>Stop F-Timer<br>Start I-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu, Serv_class=Low, Transmit=Single) | SABORT-WUPD |
| 91 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO/NO && L_sdu.len<>0 && Abort-REQ-PDU<br>=><br>Stop F-Timer<br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=FALSE, (Subnet, Instance, Reason_Code) from Abort-REQ-PDU) | WAIT-DEACT |
| 92 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=LO && L_sdu.len<>0 && NOT (Abort-REQ-PDU)<br>=><br>Reason_Code:=ABT_FE<br>Stop F-Timer<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 93 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP, DSAP=Server_SAP, Loc_add, Rem_add,L_sdu, Serv_class=Low, Update_status)<br>/Update_status=NO && L_sdu.len<>0 && NOT(Abort-REQ-PDU)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_FE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | VS-SRES |
| 94 | VS-SRES | MSAC2S_XXX.rsp(+/-)(Res_SAP)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_SE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | VS-SRES |
| 95 | VS-SRES | MSAC2S_Initiate.rsp(+/-)( Res_SAP)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER<br>Reason_Code:=ABT_SE<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | VS-SRES |
| 96 | VS-SRES | MSAC2S_Abort.req(Res_SAP, Subnet, Instance, Reason_Code)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER | VS-SRES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 97 | VS-SRES | F-Timer expired<br>=><br>Reason_Code:=ABT_TO<br>Start F-Timer<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu=Abort-REQ-PDU(Subnet=NO, Instance=MSAC2, Reason_Code), Serv_class=Low, Transmit=Single)<br>MSAC2S_Abort.ind(Res_SAP=Server_SAP, Locally_Generated=TRUE, Subnet=NO, Instance=MSAC2, Reason_Code) | SABORT-WUPD |
| 98 | ANY-STATE | unexpected DMPMS reaction<br>=><br>Stop U-Timer<br>Stop F-Timer<br>Stop I-Timer<br>MSAC2S_Fault.ind | POWER-ON |
| 99 | DEACT-PON | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status = OK | POWER-ON |
| 100 | DEACT-PON | DMPMS_SAP_DEACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC2S_Fault.ind | POWER-ON |
| 101 | any state | MSAC2S_Reset.req(Res_SAP)<br>=><br>MSAC2S_Reset.cnf(Res_SAP) | POWER-ON |

## 9.6 MSCS1S

### 9.6.1 Primitive definitions

#### 9.6.1.1 Primitives exchanged between MSCS1S and FSPMS

Table 70 shows the service primitives including their associated parameters issued by MSCS1S and received by the FSPMS.

**Table 70 – Primitives issued by MSCS1S to FSPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Set Time.ind | MSCS1S | AREP,<br>Time Value,<br>Local Time Diff,<br>Summertime,<br>Accuracy,<br>Synchronisation Active,<br>Announcement Hour | Refer to FAL Service Definition in IEC 61158-5-3 |
| Sync Interval Violation.ind | MSCS1S | AREP | |

#### 9.6.1.2 Parameter of MSCS1S primitives

The parameters used with the primitives exchanged between the FSPMS and the MSCS1S are described in FAL Service Definition in IEC 61158-5-3.

### 9.6.2 State machine description

This machine allways remains in IDLE state. It receives Clock Value indications from the DMPMS. A Set Time indication will be issued on detection of the end of a valid Clock Synchroniation sequence. Otherwise a Sync Interval Violation indication will be issued.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Local variables of the MSCS1S**

**Time Last Rcvd**
(Network Time)

This local variable contains the Clock_Value_Time_Event of the last valid received Clock Value indication.

**Error**
(Unsigned8)

This local variable is a counter for invalid Clock Synchronisation sequences. It will be reset to 0, when a Sync Interval Violation is indicated.

**Macros of the MSCS1S:**
**SET_TIME_ATTRIBUTES**
(

Local Time Diff :=CS_list.Clock_value_status.C*(-1)^CS_list.Clock_value_status.CV

Summertime := CS_list.Clock_value_status.SWT

Accuracy := CS_list.Clock_value_status.CR

Synchronisation Active := CS_list.Clock_value_status.SYF

Announcement Hour :=CS_list.Clock_value_status.ANH

)

### 9.6.3 MSCS1S state table

Table 71 contains the complete description of the MSCS1S state machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 71 – MSCS1S state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == SV && Error < 2<br>=><br>Error := Error+1 | Idle |
| 2 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE != Time Last Rcvd && Error < 2<br>=><br>Error := Error+1 | Idle |
| 3 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == SV && Error >= 2<br>=><br>Error := 0<br>Sync Interval Violation.ind | Idle |
| 4 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE != Time Last Rcvd &&Error >= 2<br>=><br>Error := 0<br>Sync Interval Violation.ind | Idle |
| 5 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE == Time Last Rcvd<br>=><br>Error := 0<br>Time Last Rcvd:= CS_list.Clock_Value_Time_Event<br>Time Value := Time Last Rcvd + Receive Delay Time<br>SET_TIME_ATTRIBUTES<br>Set_Time.ind(Time Value, Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour) | Idle |

## 9.7 MSCY1M

### 9.7.1 Primitive definitions

#### 9.7.1.1 Primitives exchanged between FSPMM1 and MSCY1M

Table 72 shows the service primitives including their associated parameters issued by the FSPMM1 and received by the MSCY1M.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 72 – Primitives issued by FSPMM1 to MSCY1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit MS0.req | FSPMM1 | Rem Add | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMM1 | (none) | |
| Abort.req | FSPMM1 | Rem Add,<br>Subnet,<br>Instance,<br>Reason Code | |
| Start Slave Handler.req | FSPMM1 | Rem Add | Start first Cycle of MSCY1M |
| Stop Slave Handler.req | FSPMM1 | Rem Add | Stop processing of MSCY1M |
| Cont Slave Handler.req | FSPMM1 | Rem Add,<br>Output Clear | Start new Cycle of MSCY1M |
| Get Slave Diag.req | FSPMM1 | Rem Add | Refer to FAL Service Definition in IEC 61158-5-3 |
| Set Output.req | FSPMM1 | Rem Add,<br>Slot_Number<br>Output Data | |
| Get Input.req | FSPMM1 | Rem Add,<br>Slot Number | |

Table 73 shows the service primitives including their associated parameters issued by the MSCY1M and received by the FSPMM1.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 73 – Primitives issued by MSCY1M to FSPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit MS0.cnf | MSCY1M | Rem Add | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | MSCY1M | (none) | |
| Start Slave Handler.cnf | MSCY1M | Rem Add | Start first Cycle of MSCY1M |
| Stop Slave Handler.cnf | MSCY1M | Rem Add | Stop processing of MSCY1M |
| Cont Slave Handler.cnf | MSCY1M | Rem Add,<br>Diag,<br>No AClr | Start new Cycle of MSCY1M |
| Get Slave Diag.cnf(+) | MSCY1M | Rem Add,<br>CREP,<br>Diag Data | Refer to FAL Service Definition in IEC 61158-5-3 |
| Get Slave Diag.cnf(-) | MSCY1M | Rem Add | |
| Set Output.cnf(+) | MSCY1M | Rem Add,<br>Slot Number | |
| Set Output.cnf(-) | MSCY1M | Rem Add,<br>Slot Number | |
| Get Input.cnf(+) | MSCY1M | Rem Add,<br>Slot_Number,<br>Input Data | |
| Get Input.cnf(-) | MSCY1M | Rem Add | |
| New Slave Diag.ind | MSCY1M | Rem Add | |
| New Input.ind | MSCY1M | Rem Add | |
| Started.ind | MSCY1M | Rem Add,<br>Alarm Limit | |
| Stopped.ind | MSCY1M | Rem Add | |
| Fault.ind | MSCY1M | Rem Add | |

#### 9.7.1.2 Parameters of MSCY1M primitives

The parameters used with the primitives exchanged between the FSPMM1 and MSCY1M are described in Table 74.

**Table 74 – Parameters used with primitives exchanged between FSPMM1 and MSCY1M**

| Parameter name | Description |
|---|---|
| Rem Add | This parameter is used to identify the DP-slave which is assigned to that communication relation. |
| Output Clear | This parameter indicates to the MSCY1M state machine that the Outputs shall be cleared. |
| Diag | This parameter indicates to the FSPMM1 state machine that no Data Exchange was processed in the preceeding cycle or the Slave was deactivated. |
| No Aclr | This parameter indicates to the FSPMM1 state machine that the corresponding DP-slave is operated in the non Autoclear-Mode (this DP-slave has no influence to the Masters State (OPERATE or CLEAR). |

Other parameters used with the primitives exchanged between the FSPMM1 and the MSCY1M are described in FAL Service Definition in IEC 61158-5-3.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 9.7.2 State machine description

For each possible DP-slave a State Machine MSCY1M (Slave-Handler) is established and started by a superior State Machine, the FSPMM1. The remote address of the Slave (Rem_Add) is used as a local reference for MSCY1M.

The Slave-Handler manages the individual states for every DP-slave. The following main states will be distinguished:

– diagnostic,
– parameterisation,
– configuration,
– data exchange.

**SI_Para_Exist**
Access: -r

Local information indicating whether or not a Slave parameter set of the associated Rem_Add exists.

**SPara**
Access:-r/-w

Any Information from ARL/CRL related to this DP-slave.

**BOutput**
Access: -r

Local storage of Output data for the DP-slave.

**BInput**
Access: -w

Local storage of Input data for the DP-slave.

**BDiag**
Access: -r/-w

Local storage of Diagnostic information of the DP-slave.

MSCY1M changes the following flags of the diagnostic information:

Invalid_Slave_Response

Station_Non_Existent

Deactivated

**DTrans**
This variable is mapped on the Data Transfer List attribute of CR-List.

Access: –r/-w

The user data exchange mode between the DP-master (Class 1) and its assigned DP-slaves is supervised on the Master side. This means it is checked whether an user data transfer to the associated DP-slaves was executed within the last DP cycle or a diagnostic cycle, after which the directly following Slave state is again the data transfer state. Any other following state forces the associated bit of the Slave to be cleared.

**SDiag**
This variable is mapped on the System Diagnosis List attribute of CR-List.

Access: -r/-w

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

It is set while the MSCY1M state machine of an activated Slave is not able to establish the connection to the Slave without errors. E. g. it is set if a Slave does not respond, if he responds with a negative DL confirmation or if he does not respond with the correct number of input data during Data_Exchange. The bit is reset after the connection could be established without any error. It is also reset if the Slave is deactivated by the User or if the MSCY1M State Machine is stopped

**Local variables**

**Delay_Count**
(Unsigned8)

The Delay_Count is used to count the number of Slave_Diag.cnf in the state DIAG2 while Diag_Data.Prm_Req is still set (Slow Slave). The Delay_Count is initialised with the value of SPara.Diag_Upd_Delay.

Range: 0 to 15 (extendible up to 255)

**MSAL1M_Started**
(Boolean)

This variable is used to store the information whether the MSAL1M State Machine is started (True) or not (False).

**Wait_for_Start_con**
(Boolean)

This variable is used to store the information whether a Start.req has not yet been confirmed by the MSAL1M State Machine (True) or not (False).

**Go-Abort**
(Boolean)

This variable is used to store the information that an error occurred forcing the MSCY1M State Machine to abort the connection.

**Macros**

**CHANGE_DIAG_IND**
(

MSCY1M_New Slave Diag.ind(Rem_Add)

if (MSAL1M_Started = TRUE)

MSAL1M_Change Diag.req(Rem_Add)

)

**GO_ABORT**
(

DTrans = 0

Go_Abort = TRUE

BInp = Nil

)

**START_MSAL1M**
(

if ((SPara.DPV1_Supported= TRUE)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

&& (Wait_for_Start_con = FALSE)

&& (MSAL1M_Started = FALSE))

MSAL1M_Start.req(Rem_Add),

MSAL1M_Started = TRUE

)

### 9.7.3 MSCY1M state table

Table 75 contains the complete description of the MSCY1M state machine.

**Table 75 – MSCY1M state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | MSCY1M MInit MS0.req ( Rem Add )<br>=><br>BDiag := NIL, SDiag := 0, DTrans := 0<br>Go_Abort := FALSE<br>MSAL1M_Started := FALSE<br>ExtPrmBsy:= FALSE<br>MSCY1M MInit MS0.cnf( Rem Add ) | STOP |
| 2 | STOP | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | STOP |
| 3 | STOP | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | STOP |
| 4 | STOP | MSCY1M Start Slave Handler.req ( Rem Add )<br>/SI_Para_Exist = FALSE \|\| SPara.Active = FALSE<br>=><br>MSCY1M Start Slave Handler.cnf( Rem Add ) | DEACT |
| 5 | STOP | MSCY1M Start Slave Handler.req ( Rem Add )<br>/SI_Para_Exist = TRUE && SPara.Active = TRUE<br>=><br>SDiag := 1, BInput := Nil, BOutput := Nil<br>BDiag.Deactivated := FALSE<br>BDiag.Station_Non_Existent := TRUE<br>MSCY1M Start Slave Handler.cnf( Rem Add ) | DIAG1 |
| 6 | STOP | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>MSCY1M Get Slave Diag.cnf(-) ( Rem Add ) | STOP |
| 7 | STOP | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>MSCY1M Set Output.cnf(-) ( Rem Add, Slot Number ) | STOP |
| 8 | STOP | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>MSCY1M Get Input.cnf(-) ( Rem Add ) | STOP |
| 9 | DEACT | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | DEACT |
| 10 | DEACT | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | DEACT |
| 11 | DEACT | MSCY1M Stop Slave Handler.req ( Rem Add )<br>=><br>MSCY1M Stop Slave Handler.cnf( Rem Add ) | STOP |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 12 | DEACT | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/SI_Para_Exist = FALSE \|\| SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DEACT |
| 13 | DEACT | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/SI_Para_Exist = TRUE && SPara.Active = TRUE<br>=><br>SDiag := 1, BInput := Nil, BOutput := Nil<br>BDiag.Deactivated := FALSE<br>BDiag.Station_Non_Existent := TRUE<br>DMPMM1 Slave Diag.req ( Rem Add ) | WDIAG1 |
| 14 | DEACT | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>MSCY1M Get Slave Diag.cnf(-) ( Rem Add ) | DEACT |
| 15 | DEACT | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>MSCY1M Set Output.cnf(-) ( Rem Add, Slot Number ) | DEACT |
| 16 | DEACT | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>MSCY1M Get Input.cnf(-) ( Rem Add ) | DEACT |
| 17 | DIAG1 | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | DIAG1 |
| 18 | DIAG1 | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | DIAG1 |
| 19 | DIAG1 | MSCY1M Stop Slave Handler.req ( Rem Add )<br>=><br>BDiag.Deactivated := TRUE, SDiag := 0<br>MSCY1M Stop Slave Handler.cnf( Rem Add ) | STOP |
| 20 | DIAG1 | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>=><br>ExtPrmBsy:=FALSE<br>DMPMM1 Slave Diag.req ( Rem Add ) | WDIAG1 |
| 21 | DIAG1 | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | DIAG1 |
| 22 | DIAG1 | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | DIAG1 |
| 23 | DIAG1 | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | DIAG1 |
| 24 | WDIAG1 | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | WDIAG1 |
| 25 | WDIAG1 | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | WDIAG1 |
| 26 | WDIAG1 | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | WDIAG1 |
| 27 | WDIAG1 | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | WDIAG1 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 28 | WDIAG1 | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | WDIAG1 |
| 29 | WDIAG1 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>BDiag.Deactivated := TRUE, SDiag := 0<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DEACT |
| 30 | WDIAG1 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>BDiag.Deactivated := TRUE, SDiag := 0<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DEACT |
| 31 | WDIAG1 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status=NA<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Station_Non_Existent := TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 32 | WDIAG1 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status=RE/RS/RR/UE/NR<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Invalid_Slave_Response := TRUE<br>BDiag.Station_Non_Existent := FALSE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 33 | WDIAG1 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status=DS<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 34 | WDIAG1 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && Diag_Data.Master_Lock = TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag := Diag_Data<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 35 | WDIAG1 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && Diag_Data.Master_Lock = FALSE<br>&& Diag_Data.Master_Add <> invalid && SPara.Prm_Data.DPV1_Enable=TRUE &&<br>SPara.DPV1_Supported=TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag := Diag_Data<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | UNLCK |
| 36 | WDIAG1 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && Diag_Data.Master_Lock = FALSE<br>&& ( Diag_Data.Master_Add = invalid \|\| SPara.Prm_Data.DPV1_Enable=FALSE \|\|<br>SPara.DPV1_Supported=FALSE)<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>SPara.New_Prm := FALSE<br>BDiag := Diag_Data, CHANGE_DIAG_IND<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | PRM |
| 37 | PRM | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | PRM |
| 38 | PRM | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 39 | PRM | MSCY1M Stop Slave Handler.req ( Rem Add )<br>=><br>BDiag.Deactivated := TRUE, SDiag := 0<br>MSCY1M Stop Slave Handler.cnf( Rem Add ) | STOP |
| 40 | PRM | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/ ExtPrmBsy=FALSE<br>=><br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Lock_Req := TRUE<br>Prm_Data.Unlock_Req := FALSE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | WPRM |
| 41 | PRM | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | PRM |
| 42 | PRM | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | PRM |
| 43 | PRM | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | PRM |
| 44 | WPRM | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | WPRM |
| 45 | WPRM | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | WPRM |
| 46 | WPRM | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | WPRM |
| 47 | WPRM | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | WPRM |
| 48 | WPRM | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | WPRM |
| 49 | WPRM | DMPMM1 Set Prm.cnf(+) ( Rem Add )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | UNLCK |
| 50 | WPRM | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | UNLCK |
| 51 | WPRM | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = DS<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | PRM |
| 52 | WPRM | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = NA<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Station_Non_Existent := TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 53 | WPRM | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = RE/RS/RR/UE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Invalid_Slave_Response := TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 54 | WPRM | DMPMM1 Set Prm.cnf(+) ( Rem Add )<br>/SPara.Active = TRUE && ExtPrmFlag = FALSE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | CFG |
| 55 | CFG | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | CFG |
| 56 | CFG | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | CFG |
| 57 | CFG | MSCY1M Stop Slave Handler.req ( Rem Add )<br>=><br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | SUNLCK |
| 58 | CFG | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>=><br>Cfg_Data:=SPara .Cfg_Data<br>DMPMM1 Chk Cfg.req ( Rem Add, Cfg Data ) | WCFG |
| 59 | CFG | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | CFG |
| 60 | CFG | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | CFG |
| 61 | CFG | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | CFG |
| 62 | WCFG | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | WCFG |
| 63 | WCFG | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | WCFG |
| 64 | WCFG | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | WCFG |
| 65 | WCFG | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | WCFG |
| 66 | WCFG | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | WCFG |
| 67 | WCFG | DMPMM1 Chk Cfg.cnf(+) ( Rem Add )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | UNLCK |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 68 | WCFG | DMPMM1 Chk Cfg.cnf(-) ( Rem Add, Status )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | UNLCK |
| 69 | WCFG | DMPMM1 Chk Cfg.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = DS<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | CFG |
| 70 | WCFG | DMPMM1 Chk Cfg.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = NA<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Station_Non_Existent := TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 71 | WCFG | DMPMM1 Chk Cfg.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = RE/RS/RR/UE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Invalid_Slave_Response := TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 72 | WCFG | DMPMM1 Chk Cfg.cnf(+) ( Rem Add )<br>/SPara.Active = TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSAL1M_Started := FALSE<br>Delay_Count := SPara .Diag_Upd_Delay<br>Wait_for_Start_con := FALSE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG2 |
| 73 | DIAG2 | MSCY1M Abort.req ( Rem Add )<br>=><br>GO_ABORT | DIAG2 |
| 74 | DIAG2 | MSAL1 Start.cnf (Rem Add)<br>=><br>Wait_for_Start_con:=FALSE<br>MSCY1M_Started.Ind(Rem_Add) | DIAG2 |
| 75 | DIAG2 | MSAL1 Abort.ind ( Rem_Add )<br>=><br>GO_ABORT | DIAG2 |
| 76 | DIAG2 | MSCY1M Stop Slave Handler.req ( Rem Add )<br>/SPara.DPV1_Supported = FALSE<br>=><br>Go_Abort := FALSE, BInput := Nil<br>DTrans := 0<br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | SUNLCK |
| 77 | DIAG2 | MSCY1M Stop Slave Handler.req ( Rem Add )<br>/SPara.DPV1_Supported = TRUE<br>=><br>Go_Abort := FALSE, BInput := Nil<br>DTrans := 0<br>MSAL1 Stop.req (Rem Add) | WSTPCS |
| 78 | DIAG2 | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort = FALSE && Wait_for_Start_con = FALSE<br>=><br>DMPMM1 Slave Diag.req ( Rem Add ) | WDIAG2 |
| 79 | DIAG2 | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort = FALSE && Wait_for_Start_con = TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG2 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 80 | DIAG2 | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort=TRUE && SPara.DPV1_Supported = FALSE<br>=><br>Go_Abort := FALSE<br>No_AClr := SPara_.Ignore_AClr OR NOT SPara.Active<br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | WUNLCK |
| 81 | DIAG2 | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort=TRUE && SPara.DPV1_Supported = TRUE<br>=><br>Go_Abort := FALSE, Diag := TRUE<br>No_AClr := SPara_.Ignore_AClr OR NOT SPara.Active<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr )<br>MSAL1 Stop.req (Rem Add) | WSTPCC |
| 82 | DIAG2 | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | DIAG2 |
| 83 | DIAG2 | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | DIAG2 |
| 84 | DIAG2 | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | DIAG2 |
| 85 | WDIAG2 | MSCY1M Abort.req ( Rem Add )<br>=><br>GO_ABORT | WDIAG2 |
| 86 | WDIAG2 | MSAL1 Abort.ind ( Rem_Add )<br>=><br>GO_ABORT | WDIAG2 |
| 87 | WDIAG2 | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | WDIAG2 |
| 88 | WDIAG2 | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | WDIAG2 |
| 89 | WDIAG2 | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | WDIAG2 |
| 90 | WDIAG2 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=FALSE<br>BDiag := Diag_Data, GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG2 |
| 91 | WDIAG2 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = FALSE<br>=><br>GO_ABORT, Diag:=FALSE, No_AClr:=FALSE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG2 |
| 92 | WDIAG2 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = DS<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG2 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 93 | WDIAG2 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = NA && SPara_.NA_To_Abort = FALSE<br>=><br>Diag:=TRUE,No_ACIr:=SPara_.Ignore_ACIr<br>BDiag.Station_Non_Existent := TRUE<br>SDiag := 1, DTrans := 0<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No ACIr ) | DIAG2 |
| 94 | WDIAG2 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = NA && SPara_.NA_To_Abort = TRUE<br>=><br>Diag:=TRUE, No_ACIr:=SPara_.Ignore_ACIr, BDiag.Station_Non_Existent := TRUE,<br>SDiag := 1, GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No ACIr ) | DIAG2 |
| 95 | WDIAG2 | DMPMM1 Slave Diag.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = RE/RS/RR/UE/NR<br>=><br>Diag:=TRUE, No_ACIr:=SPara_.Ignore_ACIr, BDiag.Station_Non_Existent := TRUE,<br>BDiag.Invalid_Slave_Response := TRUE, SDiag := 1, GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No ACIr ) | DIAG2 |
| 96 | WDIAG2 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && (Diag_Data.Prm_Req = TRUE \|\| Diag_Data.Master_Lock = TRUE) && Delay_Count > 0<br>=><br>Diag:=TRUE, No_ACIr:=SPara_.Ignore_ACIr<br>Delay_Count := Delay_Count-1, BDiag := Diag_Data<br>BInput := 0, SDiag := 1, DTrans := 0<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No ACIr ) | DIAG2 |
| 97 | WDIAG2 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && (Diag_Data.Prm_Req = TRUE \|\| Diag_Data.Master_Lock = TRUE) && Delay_Count = 0<br>=><br>Diag:=TRUE, No_ACIr:=SPara_.Ignore_ACIr<br>BDiag := Diag_Data, SDiag := 1, GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No ACIr ) | DIAG2 |
| 98 | WDIAG2 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && Diag_Data.Prm_Req = FALSE &&<br>Diag_Data.Master_Lock = FALSE && Diag_Data.Station_Not_Ready = TRUE<br>=><br>Diag:=TRUE, No_ACIr:=SPara_.Ignore_ACIr<br>SDiag := 1, DTrans := 0, BInput := Nil<br>BDiag := Diag_Data<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No ACIr ) | DIAG2 |
| 99 | WDIAG2 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && Diag_Data.Prm_Req = FALSE &&<br>Diag_Data.Master_Lock = FALSE && Diag_Data.Station_Not_Ready = FALSE &&<br>Diag_Data.Stat_Diag = TRUE<br>=><br>Diag:=TRUE, No_ACIr:=SPara_.Ignore_ACIr<br>START_MSAL1M, SDiag := 1, DTrans := 0<br>BInput := Nil, BDiag := Diag_Data<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No ACIr ) | DIAG2 |
| 100 | WDIAG2 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && Diag_Data.Prm_Req = FALSE &&<br>Diag_Data.Master_Lock = FALSE && Diag_Data.Station_Not_Ready = FALSE &&<br>Diag_Data.Stat_Diag=FALSE && DTrans = 0<br>=><br>Diag:=TRUE, No_ACIr:=TRUE<br>SDiag := Diag_Data.Ext_Diag, Delay_Count := 0<br>BDiag := Diag_Data, START_MSAL1M,<br>CHANGE_DIAG_IND<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No ACIr ) | DATA |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 101 | WDIAG2 | DMPMM1 Slave Diag.cnf(+) ( Rem Add, Diag Data )<br>/SPara.Active = TRUE && Diag_Data.Prm_Req = FALSE &&<br>Diag_Data.Master_Lock = FALSE && Diag_Data.Station_Not_Ready = FALSE &&<br>Diag_Data.Stat_Diag = FALSE && DTrans = 1<br>=><br>Diag:=TRUE, No_AClr:=TRUE<br>SDIAG := Diag_Data.Ext_Diag, CHANGE_DIAG_IND<br>BDiag := Diag_Data<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 102 | DATA | MSCY1M Abort.req ( Rem Add )<br>=><br>GO_ABORT | DATA |
| 103 | DATA | MSAL1 Start.cnf (Rem Add)<br>=><br>Wait_for_Start_con := FALSE, START_MSAL1M<br>MSCY1M_Started.ind(Rem_Add) | DATA |
| 104 | DATA | MSAL1 Abort.ind ( Rem_Add )<br>=><br>GO_ABORT | DATA |
| 105 | DATA | MSCY1M Stop Slave Handler.req ( Rem Add )<br>/SPara.DPV1_Supported = FALSE<br>=><br>Go_Abort := FALSE, BInput := Nil<br>DTrans := 0<br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | SUNLCK |
| 106 | DATA | MSCY1M Stop Slave Handler.req ( Rem Add )<br>/SPara.DPV1_Supported = TRUE<br>=><br>Go_Abort := FALSE, BInput :=Nil, DTrans := 0<br>MSAL1 Stop.req (Rem Add) | WSTPCS |
| 107 | DATA | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort = FALSE && Wait_for_Start_con = FALSE && Output_Clear = TRUE &&<br>((SPara.DPV1_Supported = FALSE && SPara.New_Prm = FALSE) ||<br>SPara.DPV1_Supported = TRUE) && SPara.Fail_Safe = FALSE<br>=><br>Outp_Data:=0<br>DMPMM1 Data Exchange.req ( Rem Add, Outp Data ) | WDATA |
| 108 | DATA | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort = FALSE && Wait_for_Start_con = FALSE && Output_Clear = TRUE &&<br>((SPara.DPV1_Supported = FALSE && SPara.New_Prm = FALSE) ||<br>SPara.DPV1_Supported = TRUE) && SPara.Fail_Safe = TRUE<br>=><br>Outp_Data.len:=0<br>DMPMM1 Data Exchange.req ( Rem Add, Outp Data ) | WDATA |
| 109 | DATA | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort = FALSE && Wait_for_Start_con = FALSE && Output_Clear = FALSE &&<br>((SPara.DPV1_Supported = FALSE && SPara.New_Prm = FALSE) ||<br>SPara.DPV1_Supported = TRUE)<br>=><br>Outp_Data:=BOutput<br>DMPMM1 Data Exchange.req ( Rem Add, Outp Data ) | WDATA |
| 110 | DATA | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort = FALSE && SPara.DPV1_Supported = FALSE && SPara.New_Prm = TRUE<br>=><br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Lock_Req := TRUE<br>Prm_Data.Unlock_Req := FALSE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | WDATA |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 111 | DATA | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort = FALSE && Wait_for_Start_con = TRUE<br>=><br>Diag:=TRUE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 112 | DATA | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort=TRUE && (SPara.DPV1_Supported = FALSE\|\|<br>SPara_PrmCmd_Supported = TRUE)<br>=><br>Go_Abort := FALSE, Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | WUNLCK |
| 113 | DATA | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/Go_Abort=TRUE && SPara.DPV1_Supported = TRUE &&<br>SPara_PrmCmd_Supported = FALSE<br>=><br>Go_Abort := FALSE, Diag := TRUE<br>No_AClr := SPara_.Ignore_AClr OR NOT SPara.Active<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr )<br>MSAL1 Stop.req (Rem Add) | WSTPCC |
| 114 | DATA | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | DATA |
| 115 | DATA | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | DATA |
| 116 | DATA | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | DATA |
| 117 | WDATA | MSCY1M Abort.req ( Rem Add )<br>=><br>GO_ABORT | WDATA |
| 118 | WDATA | MSAL1 Abort.ind ( Rem_Add )<br>=><br>GO_ABORT | WDATA |
| 119 | WDATA | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | WDATA |
| 120 | WDATA | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | WDATA |
| 121 | WDATA | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | WDATA |
| 122 | WDATA | DMPMM1 Data Exchange.cnf(+) ( Rem Add, Diag Flag, Inp Data )<br>/SPara.Active = FALSE<br>=><br>GO_ABORT, Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 123 | WDATA | DMPMM1 Data Exchange.cnf(-) ( Rem Add, Status )<br>/SPara.Active = FALSE<br>=><br>GO_ABORT, Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 124 | WDATA | DMPMM1 Data Exchange.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = DS<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>DTrans := 0, BInput := Nil<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 125 | WDATA | DMPMM1 Data Exchange.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = NA && SPara_.NA_To_Abort = FALSE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Station_Non_Existent := TRUE, SDiag := 1<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 126 | WDATA | DMPMM1 Data Exchange.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status=NA && SPara_.NA_To_Abort = TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Station_Non_Existent := TRUE, SDiag := 1<br>GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 127 | WDATA | DMPMM1 Data Exchange.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = RE/RS/RR/UE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Invalid_Slave_Response := TRUE<br>BDiag.Station_Non_Existent := FALSE<br>SDiag := 1, GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 128 | WDATA | DMPMM1 Data Exchange.cnf(+) ( Rem Add, Diag Flag, Inp Data )<br>/SPara.Active = TRUE && Diag_Flag = FALSE && Inp_Data.len = Exp_Inp_Len<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>BDiag.Station_Non_Existent := FALSE<br>BInput := Inp_Data, DTrans := 1<br>SDiag := BDiag.Ext_Diag<br>MSCY1M_New_Input.ind( Rem_Add )<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 129 | WDATA | DMPMM1 Data Exchange.cnf(+) ( Rem Add, Diag Flag, Inp Data )<br>/SPara.Active = TRUE && Inp_Data.len <> Exp_Inp_Len && (Inp_Data.len <> 1 ‖ Exp_Inp_Len <> 0 ‖ Diag_Flag = FALSE)<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Invalid_Slave_Response := TRUE<br>BDiag.Station_Non_Existent := FALSE<br>SDiag := 1, GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 130 | WDATA | DMPMM1 Data Exchange.cnf(+) ( Rem Add, Diag Flag, Inp Data )<br>/SPara.Active = TRUE && Diag_Flag = TRUE && Inp_Data.len = Exp_Inp_Len<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>BDiag.Station_Non_Existent := FALSE<br>BInput := Inp_Data, DTrans := 1<br>SDiag := BDiag.Ext_Diag<br>MSCY1M_New_Input.ind( Rem_Add )<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG2 |
| 131 | WDATA | DMPMM1 Data Exchange.cnf(+) ( Rem Add, Diag Flag, Inp Data )<br>/SPara.Active = TRUE && Diag_Flag = TRUE && Inp_Data.len = 1 && Exp_Inp_Len = 0<br>=><br>Diag:=FALSE, No_AClr:=TRUE<br>BDiag.Station_Non_Existent := FALSE, DTrans := 1<br>SDiag := BDiag.Ext_Diag<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG2 |
| 132 | WDATA | DMPMM1 Set Prm.cnf(+) ( Rem Add )<br>/SPara.Active = FALSE<br>=><br>GO_ABORT, Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 133 | WDATA | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = FALSE<br>=><br>GO_ABORT, Diag:=FALSE, No_AClr:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 134 | WDATA | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = DS<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 135 | WDATA | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = NA && SPara_.NA_To_Abort = FALSE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Station_Non_Existent := TRUE, SDiag := 1<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 136 | WDATA | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = NA && SPara_.NA_To_Abort = TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Station_Non_Existent := TRUE, SDiag := 1<br>GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 137 | WDATA | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = RE/RS/RP/UE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Invalid_Slave_Response := TRUE<br>BDiag.Station_Non_Existent := FALSE<br>SDiag := 1, GO_ABORT<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 138 | WDATA | DMPMM1 Set Prm.cnf(+) ( Rem Add )<br>/SPara.Active = TRUE<br>=><br>Diag:=TRUE, No_AClr:=TRUE<br>SPara.New_Prm := FALSE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DATA |
| 139 | UNLCK | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | UNLCK |
| 140 | UNLCK | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | UNLCK |
| 141 | UNLCK | MSCY1M Stop Slave Handler.req ( Rem Add )<br>=><br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | SUNLCK |
| 142 | UNLCK | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>=><br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | WUNLCK |
| 143 | UNLCK | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | UNLCK |
| 144 | UNLCK | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | UNLCK |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 145 | UNLCK | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | UNLCK |
| 146 | WUNLCK | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | WUNLCK |
| 147 | WUNLCK | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | WUNLCK |
| 148 | WUNLCK | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | WUNLCK |
| 149 | WUNLCK | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | WUNLCK |
| 150 | WUNLCK | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | WUNLCK |
| 151 | WUNLCK | DMPMM1 Set Prm.cnf(+) ( Rem Add )<br>/SPara.Active = TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 152 | WUNLCK | DMPMM1 Set Prm.cnf(+) ( Rem Add )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE, SDiag := 0<br>BDiag.Deactivated := TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DEACT |
| 153 | WUNLCK | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 154 | WUNLCK | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE, SDiag := 0<br>BDiag.Deactivated := TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DEACT |
| 155 | SUNLCK | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | SUNLCK |
| 156 | SUNLCK | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | SUNLCK |
| 157 | SUNLCK | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | SUNLCK |
| 158 | SUNLCK | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | SUNLCK |
| 159 | SUNLCK | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | SUNLCK |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 160 | SUNLCK | DMPMM1 Set Prm.cnf(+) ( Rem Add )<br>=><br>SDiag := 0, BDiag.Deactivated := TRUE<br>MSCY1M Stop Slave Handler.cnf( Rem Add ) | STOP |
| 161 | SUNLCK | DMPMM1 Set Prm.cnf(-) ( Rem Add, Status )<br>=><br>SDiag := 0, BDiag.Deactivated := TRUE<br>MSCY1M Stop Slave Handler.cnf( Rem Add ) | STOP |
| 162 | WSTPCC | MSAL1 Start.cnf (Rem Add)<br>=><br>MSAL1M_Started := FALSE | WSTPCC |
| 163 | WSTPCC | MSAL1 Stop.cnf (Rem Add)<br>/MSAL1M_Started = TRUE<br>=><br>MSCY1M Stopped.ind ( Rem Add ) | UNLCK |
| 164 | WSTPCC | MSAL1 Stop.cnf (Rem Add)<br>/MSAL1M_Started = FALSE<br>=><br>ignore | UNLCK |
| 165 | WSTPCC | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr OR NOT SPara.Active<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | WSTPCC |
| 166 | WSTPCC | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | WSTPCC |
| 167 | WSTPCC | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | WSTPCC |
| 168 | WSTPCC | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | WSTPCC |
| 169 | WSTPCC | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | WSTPCC |
| 170 | WSTPCC | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | WSTPCC |
| 171 | WSTPCS | MSAL1 Start.cnf (Rem Add)<br>=><br>MSAL1M_Started := FALSE | WSTPCS |
| 172 | WSTPCS | MSAL1 Stop.cnf (Rem Add)<br>/MSAL1M_Started = TRUE<br>=><br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>MSCY1M Stopped.ind ( Rem Add )<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | SUNLCK |
| 173 | WSTPCS | MSAL1 Stop.cnf (Rem Add)<br>/MSAL1M_Started = FALSE<br>=><br>Prm_Data := SPara .Prm_Data<br>Prm_Data.Unlock_Req := TRUE<br>DMPMM1 Set Prm.req ( Rem Add, Prm Data ) | SUNLCK |
| 174 | WSTPCS | MSCY1M Abort.req ( Rem Add )<br>=><br>ignore | WSTPCS |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 175 | WSTPCS | MSAL1 Abort.ind ( Rem_Add )<br>=><br>ignore | WSTPCS |
| 176 | WSTPCS | MSCY1M Get Slave Diag.req ( Rem Add )<br>=><br>Diag_Data := BDiag<br>MSCY1M Get Slave Diag.cnf(+) ( Rem Add, Diag Data ) | WSTPCS |
| 177 | WSTPCS | MSCY1M Set Output.req ( Rem Add, Slot Number, Output Data )<br>=><br>BOutput(Slot_Number) := Output_Data<br>MSCY1M Set Output.cnf(+) ( Rem Add, Slot Number ) | WSTPCS |
| 178 | WSTPCS | MSCY1M Get Input.req ( Rem Add, Slot_Number )<br>=><br>Input_Data := BInput(Slot_Number)<br>MSCY1M Get Input.cnf(+) ( Rem Add, Input Data ) | WSTPCS |
| 179 | ANY-STATE | MSCY1M Reset.req ( Rem Add )<br>=><br>MSCY1M Reset.cnf ( Rem Add ) | POWER-ON |
| 180 | WPRM | DMPMM1 Set Prm.cnf(+) ( Rem Add )<br>/SPara.Active = TRUE && ExtPrmFlag = TRUE<br>=><br>ExtPrmBsy:=TRUE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | PRM |
| 181 | PRM | MSCY1M Cont Slave Handler.req ( Rem Add, Output Clear )<br>/ ExtPrmBsy=TRUE<br>=><br>Prm_Data := SPara .ExtPrm_Data<br>DMPMM1 Set ExtPrm.req ( Rem Add, Prm Data ) | WPRM |
| 182 | WPRM | DMPMM1 Set ExtPrm.cnf(+) ( Rem Add )<br>/SPara.Active = TRUE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr, ExtPrmBsy:=FALSE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | CFG |
| 183 | WPRM | DMPMM1 Set ExtPrm.cnf(+) ( Rem Add )<br>/SPara.Active = FALSE<br>=><br>Diag:=FALSE, No_AClr:=TRUE, ExtPrmBsy:=FALSE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | UNLCK |
| 184 | WPRM | DMPMM1 Set ExtPrm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = DS<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | PRM |
| 185 | WPRM | DMPMM1 Set ExtPrm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = NA<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Station_Non_Existent := TRUE, ExtPrmBsy:=FALSE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |
| 186 | WPRM | DMPMM1 Set ExtPrm.cnf(-) ( Rem Add, Status )<br>/SPara.Active = TRUE && Status = RE/RS/RR/UE<br>=><br>Diag:=TRUE, No_AClr:=SPara_.Ignore_AClr<br>BDiag.Invalid_Slave_Response := TRUE, ExtPrmBsy:=FALSE<br>MSCY1M Cont Slave Handler.cnf( Rem Add, Diag, No AClr ) | DIAG1 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

## 9.8 MSAL1M

### 9.8.1 Primitive definitions

#### 9.8.1.1 Primitives exchanged between FSPMM1 and MSAL1M

Table 76 shows the service primitives including their associated parameters issued by the FSPMM1 and received by the MSAL1M.

**Table 76 – Primitives issued by FSPMM1 to MSAL1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit.req | FSPMM1 | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMM1 | (none) | — |
| Abort.req | FSPMM1 | (none) | — |

Table 77 shows the service primitives including their associated parameters issued by the MSAL1M and received by the FSPMM1.

**Table 77 – Primitives issued by MSAL1M to FSPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit.cnf | MSAL1M | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | MSAL1M | (none) | — |
| Fault.ind | MSAL1M | (none) | — |
| Started.ind | MSAL1M | Alarm Limit | — |
| Stopped.ind | MSAL1M | (none) | — |
| Alarm Notification.ind | MSAL1M | Slot Number,<br>Alarm Type,<br>Seq Nr,<br>Add Ack,<br>Alarm Specifier,<br>Alarm Data | — |

#### 9.8.1.2 Primitives exchanged between MSCY1M and MSAL1M

Table 78 shows the service primitives including their associated parameters issued by the MSCY1M and received by the MSAL1M.

**Table 78 – Primitives issued by MSCY1M to MSAL1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Start.req | MSCY1M | Rem Add | Start MS1-processing |
| Stop.req | MSCY1M | Rem Add | Stop MS1-processing |
| Change Diag.req | MSCY1M | Rem Add,<br>Diag Data | Indicates change of diagnosis |

Table 79 shows the service primitives including their associated parameters issued by the MSAL1M and received by the MSCY1M.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 79 – Primitives issued by MSAL1M to MSCY1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Abort.ind | MSAL1M | Rem Add | MSAL1 detected an error |
| Start.cnf | MSAL1M | Rem Add | Start completed |
| Stop.cnf | MSAL1M | Rem Add | Stop completed |

#### 9.8.1.3 Parameters of MSAL1M primitives

The parameters used with the primitives exchanged between the MSAL1M and MSCY1M are described in Table 80.

**Table 80 – Parameter used with primitives exchanged between MSAL1M and MSCY1M**

| Parameter name | Description |
|---|---|
| Rem Add | This parameter conveys the DL address of the assigned DP-slave |
| Diag Data | This parameter conveys the new diagnosis information of a DP-slave |

### 9.8.2 State machine description

The Alarm State Machine of a DP-master (Class 1) handles alarm messages, which have been indicated by a DP-slave.

When started the State Machine waits for being started by MSCY1M. The Start.req follows, when the assigned DP-slave has reached the state DATA-EXCH. MSAL1M will be stopped by MSCY1M if the corresponding DP-slave leaves the state DATA-EXCH.

The service Slave_Diag.cnf provides new Diag_Data to MSCY1M. MSCY1M copies the diagnosis information into the User data interface. Additionally MSCY1M hands over an Alarm in Ext_Diag_Data to MSAL1M.

Alarm-Requests are indicated to the User. The User has to confirm the Alarm-Indications with the service Alarm_Ack.

The MSAC1 State Machine indicates the termination of the alarm sequence with the service primitive Alarm_Ack.cnf.

Two types of alarm handling are provided by a DP-master (Class 1) the type mode and the sequence mode. Depending on the chosen mode the number of oustanding alarms per DP-slave differs. In type mode one pending alarm per type and per DP-slave is allowed. 6 different alarm types are defined. In sequence mode the alarm type is irrelevant. Only the number of simultaneously processed alarms for each DP-slave is limited to a value between 2 and 32.

**Local variables**

**Alarm_Sequence**
(Boolean)

This variable indicates whether:

only one alarm of a specific alarm type can be active at one time (type mode: Alarm_Sequence=False) or

several alarms (2 to 32) of any type can be active at one time (sequence mode: Alarm_Sequence=True).

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Alarm_State_Table**
(Unsigned8)

The Alarm_State_Table is a two dimensional array with 7 * 32 elements. Each element within the array stores information about the actual state of any alarm. The array is indexed with the alarm class calculated from the Alarm_Type and the Seq_Nr of the alarm.

**Range**
Table 81 shows the possible values for the actual states of alarms in the Alarm_State_Table.

**Table 81 – Possible values in the Alarm_State_Table**

| Value | Meaning |
|---|---|
| idle | No service is being processed according to the actual index. |
| w_res | Wait for MSAL1_Alarm.res according to the actual index. |
| w_req | For this index (Alarm_Type,Seq_Nr) an Alarm_Acknowledge is stored in the Alarm_Ack_FIFO and MSAL1M waits for another MSAC1M_Alarm_Ack.cnf. |
| w_con | Wait for MSAC1M_Alarm_Ack.cnf according to the actual index. |

**Alarm_Limit**
(Unsigned8)

This variable indicates the maximum number of parallel alarms of each Slave.

Range: 7..Bus_Para.Alarm_Max

**Alarm_Decode**
(Array 0..7 of Unsigned8)

Table for Decoding the number of parallel alarms.

Range:

Alarm_Decode[0]= 1
Alarm_Decode[1]= 2
Alarm_Decode[2]= 4
Alarm_Decode[3]= 8
Alarm_Decode[4]=12
Alarm_Decode[5]=16
Alarm_Decode[6]=24
Alarm_Decode[7]=32

**Actual_Max_Alarm_Len**
(Unsigned8)

Actual_Max_Alarm_Len contains the actual maximum length of an Alarm.

Range: 4..64

**Alarm_Count**
(Unsigned8)

This counter contains the number of alarms which have actually been sent by the Slave. The counter is only used in the sequence mode.

Range: 0..32

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Waiting_For_Alarm_Ack**
(Boolean)

This variable shows, whether the Alarm State Machine is actually waiting for the confirmation of any alarm acknowledge (True) or not (False).

**Alarm_Ack_FIFO**
The Alarm_Ack_FIFO is used to store the Alarm_Ack which are still to be sent. Each element of this alarm acknowledge FIFO has 3 Unsigned8 entries: Alarm_Type, Slot_Number and Seq_Nr. The FIFO shall be able to hold up to 32 respectively Bus_Para.Alarm_Max entries.

**Alarm_Ack_FIFO_Filled**
(Boolean)

This variable indicates whether the Alarm_Ack_FIFO contains at least one entry (True) or it is empty (False). The value of the variable is adapted with each access to the FIFO.

**Act_Alarm_PDU**
(Octet-String)

The structure Act_Alarm_PDU is temporarily used to store one Alarm while being evaluated.

**Stored_Alarm_PDU**
(Octet-String)

The structure Stored_Alarm_PDU is used to store exactly one Alarm-PDU. Up to one Alarm-PDU shall be saved locally during waiting for MSAC1M_Alarm_Ack.cnf of the according Seq_Nr, because only one channel for alarm acknowledges exists.

**Actual_Enabled_Alarms**
(Bitarea[0..7])

This variable indicates the type of alarms which are actually supported by the Master.

Bit 0 = reserved
Bit 1 = reserved
Bit 2 = Update_Alarm
Bit 3 = Status_Alarm
Bit 4 = Manufacturer_Specific_Alarm
Bit 5 = Diagnostic_Alarm
Bit 6 = Process_Alarm
Bit 7 = Pull_Plug_Alarm

**Functions**

**Fill_Alarm_State_Table(State)**
This function sets each entry of the two dimensional Alarm_State_Table to the given initial value of State. Valid parameters for State are idle, w_res, w_req and w_con. The function has no return value.

**Reset_Alarm_Ack_FIFO()**
This function resets the FIFO for alarm acknowledges. During this function all entries are cleared an the variable Alarm_Ack_FIFO_Filled is thereby set to False. The function has no return value.

**Store_To_Alarm_Ack_FIFO(Alarm_Type, Slot_Number, Seq_Nr)**
By means of this function one set of the 3 alarm recognition parameters Alarm_Type, Slot_Number and Seq_Nr is stored from the given parameters into the Alarm_Ack_FIFO. The function has no return value.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Load_From_Alarm_Ack_FIFO(Alarm_Type, Slot_Number, Seq_Nr)**
By means of this function one set of the 3 alarm recognition parameters Alarm_Type, Slot_Number and Seq_Nr is read from the Alarm_Ack_FIFO and stored into the given parameters. The function has no return value.

**AcIs(Alarm_Type)**
This function calculates the Alarm_Type related index as return value as follows:

```
if (Alarm_Type = 1) return 0
if (Alarm_Type = 2) return 1
if (Alarm_Type = 3) return 2
if (Alarm_Type = 4) return 3
if (Alarm_Type = 5) return 4
if (Alarm_Type = 6) return 5
if (Alarm_Type >= 32) && (Alarm_Type <= 126) return 6
```

**Macros**
**ALARM_ENABLED**

```
(
((Alarm_Type=3 OR Alarm_Type=4)
&& Actual_Enabled_Alarms[7]=TRUE)
|| ((Alarm_Type=2) && Actual_Enabled_Alarms[6]=TRUE)
|| ((Alarm_Type=1) && Actual_Enabled_Alarms[5]=TRUE)
|| ((Alarm_Type>=32 && Alarm_Type<=126)
&& Actual_Enabled_Alarms[4]=TRUE)
|| ((Alarm_Type=5) && Actual_Enabled_Alarms[3]=TRUE)
|| ((Alarm_Type=6) && Actual_Enabled_Alarms[2]=TRUE)
)
```

### 9.8.3 MSAL1M state table

Table 82 contains the complete description of the MSAL1M state machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 82 – MSAL1M state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | MSAL1M_Minit_MS1.req(Rem_Add)<br>=>MSAL1M_Minit_MS1.cnf(Rem_Add) | WSTART |
| 2 | POWER-ON | MSAL1M_Abort.req(Rem_Add)<br>=>ignore | POWER-ON |
| 3 | POWER-ON | MSAL1M_Alarm_Ack.req(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr)<br>=>MSAL1M_Alarm_Ack.cnf(-)(Rem_Add, Status:=Not_Initialized) | POWER-ON |
| 4 | WSTART | MSCY1M_Start.req(Rem_Add)<br>/Alarm_Mode = 0<br>=>Alarm_Sequence := False<br>Alarm_Limit := 7<br>Actual_Enabled_Alarms := Enabled_Alarms<br>Actual_Max_Alarm_Len := Max_Alarm_Len<br>Fill_Alarm_State_Table(idle)<br>Reset_Alarm_Ack_FIFO()<br>Alarm_Count := 0<br>Clear Stored_Alarm_PDU<br>Waiting_For_Alarm_Ack := False<br>MSAL1M_Started.ind(Rem_Add, Alarm_Limit)<br>MSCY1M_Start.cnf(Rem_Add) | W-DIA-EVENT |
| 5 | WSTART | MSCY1M_Start.req(Rem_Add)<br>/Alarm_Mode <> 0<br>=>Alarm_Sequence := True<br>Alarm_Limit :=<br>min(Bus_Para.Alarm_Max, Alarm_Decode[Alarm_Mode])<br>Actual_Enabled_Alarms := Enabled_Alarms<br>Actual_Max_Alarm_Len := Max_Alarm_Len<br>Fill_Alarm_State_Table(idle)<br>Reset_Alarm_Ack_FIFO()<br>Alarm_Count := 0<br>Clear Stored_Alarm_PDU<br>Waiting_For_Alarm_Ack := False<br>MSAL1M_Started.ind(Rem_Add, Alarm_Limit)<br>MSCY1M_Start.cnf(Rem_Add) | W-DIA-EVENT |
| 6 | WSTART | MSCY1M_Stop.req(Rem_Add)<br>=>MSAL1M_Fault.ind | POWER-ON |
| 7 | WSTART | MSCY1M_Change_Diag.req(Rem_Add, Diag_Block)<br>=>MSAL1M_Fault.ind | WSTART |
| 8 | WSTART | MSAL1M_Abort.req(Rem_Add)<br>=>ignore | WSTART |
| 9 | WSTART | MSAL1M_Alarm_Ack.req(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr)<br>=>MSAL1M_Alarm_Ack.cnf(-)(Rem_Add, Status:=Not_Started) | WSTART |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 10 | WSTART | MSAC1M_Alarm_Ack.cnf(+)(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>=>MSAL1M_Fault.ind | POWER-ON |
| 11 | W-DIA-EVENT | MSAL1M_Minit_MS1.req(Rem_Add)<br>=>MSAL1M_Fault.ind | POWER-ON |
| 12 | W-DIA-EVENT | MSAL1M_Abort.req(Rem_Add)<br>=>MSAL1M_Abort.ind(Rem_Add) | WSTART |
| 13 | W-DIA-EVENT | MSCY1M_Start.req(Rem_Add)<br>=>MSAL1M_Fault.ind | POWER-ON |
| 14 | W-DIA-EVENT | MSCY1M_Stop.req(Rem_Add)<br>=>MSAL1M_Stopped.ind(Rem_Add)<br>MSCY1M_Stop.cnf(Rem_Add) | WSTART |
| 15 | W-DIA-EVENT | MSCY1M_Change_Diag.req(Rem_Add, Diag_Block)<br>=>Act_Alarm_PDU := Alarm-PDU | CHK-DIA-ALARM |
| 16 | W-DIA-EVENT | MSAL1M_Alarm_Ack.req(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr)<br>/ALARM_ENABLED(Alarm_Type) = False<br>=>MSAL1M_Alarm_Ack.cnf(-)(Rem_Add, Status:=Not_Enabled) | W-DIA-EVENT |
| 17 | W-DIA-EVENT | MSAL1M_Alarm_Ack.req(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr)<br>/ALARM_ENABLED(Alarm_Type) = True &&<br>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] <> w_res<br>=>MSAL1M_Alarm_Ack.cnf(-)(Rem_Add, Status:=Alarm_Not_Pending) | W-DIA-EVENT |
| 18 | W-DIA-EVENT | MSAL1M_Alarm_Ack.req(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr)<br>/ALARM_ENABLED(Alarm_Type) = True && Alarm_Sequence = False &&<br>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] = w_res &&<br>Waiting_For_Alarm_Ack = False<br>=>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] := w_con<br>Waiting_For_Alarm_Ack := True<br>MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr) | W-DIA-EVENT |
| 19 | W-DIA-EVENT | MSAL1M_Alarm_Ack.req(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr)<br>/ALARM_ENABLED(Alarm-PDU.Alarm_Type) = True &&<br>Alarm_Sequence = False &&<br>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] = w_res &&<br>Waiting_For_Alarm_Ack = True<br>=>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] := w_req<br>Store_To_Alarm_Ack_FIFO(Alarm_Type, Slot_Number, Seq_Nr) | W-DIA-EVENT |
| 20 | W-DIA-EVENT | MSAL1M_Alarm_Ack.req(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr)<br>/ALARM_ENABLED(Alarm_Type) = True && Alarm_Sequence = True &&<br>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] = w_res &&<br>Waiting_For_Alarm_Ack = False<br>=>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] := w_con<br>Alarm_Count := Alarm_Count-1<br>Waiting_For_Alarm_Ack := True<br>MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr) | W-DIA-EVENT |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 21 | W-DIA-EVENT | MSAL1M_Alarm_Ack.req(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr)<br>/ALARM_ENABLED(Alarm_Type) = True && Alarm_Sequence = True &&<br>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] = w_res &&<br>Waiting_For_Alarm_Ack = True<br>=>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] := w_req<br>Store_To_Alarm_Ack_FIFO(Alarm_Type, Slot_Number, Seq_Nr) | W-DIA-EVENT |
| 22 | W-DIA-EVENT | MSAC1M_Alarm_Ack.cnf(+)(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/Waiting_For_Alarm_Ack = False<br>=>MSAL1M_Fault.ind | POWER-ON |
| 23 | W-DIA-EVENT | MSAC1M_Alarm_Ack.cnf(+)(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/Waiting_For_Alarm_Ack = True &&<br>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] <> w_con<br>=>MSAL1M_Abort.ind(Rem_Add)<br>MSAL1M_Stopped.ind(Rem_Add) | WSTART |
| 24 | W-DIA-EVENT | MSAC1M_Alarm_Ack.cnf(+)(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/Waiting_For_Alarm_Ack = True &&<br>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] = w_con<br>=>Alarm_State_Table[Acls(Alarm_Type), Seq_Nr] := idle<br>Waiting_For_Alarm_Ack := False<br>MSAL1M_Alarm_Ack.cnf(+)(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr) | SEND-NEXT-ALARM |
| 25 | CHK-DIA-ALARM | /Act_Alarm_PDU not present<br>=>ignore | W-DIA-EVENT |
| 26 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = False<br>=>MSAL1M_Abort.ind(Rem_Add)<br>MSAL1M_Stopped.ind(Rem_Add) | WSTART |
| 27 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True &&<br>Alarm_Sequence = False &&<br>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] = w_res/w_req<br>=>MSAL1M_Abort.ind(Rem_Add)<br>MSAL1M_Stopped.ind(Rem_Add) | WSTART |
| 28 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True &&<br>Alarm_Sequence = False &&<br>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] = w_con &&<br>Stored_Alarm_PDU present<br>=>MSAL1M_Abort.ind(Rem_Add)<br>MSAL1M_Stopped.ind(Rem_Add) | WSTART |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 29 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True &&<br>Alarm_Sequence = False &&<br>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] = idle<br>=>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] := w_res<br>MSAL1M_Alarm_Notification.ind(Rem_Add,<br>Alarm_Type:=Act_Alarm_PDU.Alarm_Type,<br>Slot_Number:=Act_Alarm_PDU.Slot_Number,<br>Seq_Nr:=Act_Alarm_PDU.Seq_Nr,<br>Alarm_Specifier:=Act_Alarm_PDU.Alarm_Specifier,<br>Add_Ack:=Act_Alarm_PDU.Add_Ack,<br>Alarm_Data:=Act_Alarm_PDU.Diagnostic_User_Data) | W-DIA-EVENT |
| 30 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True &&<br>Alarm_Sequence = False &&<br>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] = w_con &&<br>Stored_Alarm_PDU not present<br>=>Stored_Alarm_PDU := Act_Alarm_PDU | W-DIA-EVENT |
| 31 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True &&<br>Alarm_Sequence = True && Alarm_Count >= Alarm_Limit<br>=>MSAL1M_Abort.ind(Rem_Add)<br>MSAL1M_Stopped.ind(Rem_Add) | WSTART |
| 32 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True &&<br>Alarm_Sequence = True && Alarm_Count < Alarm_Limit &&<br>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] = w_res/w_req<br>=>MSAL1M_Abort.ind(Rem_Add)<br>MSAL1M_Stopped.ind(Rem_Add) | WSTART |
| 33 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True &&<br>Alarm_Sequence = True && Alarm_Count < Alarm_Limit &&<br>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] = idle<br>=>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] := w_res<br>Alarm_Count := Alarm_Count+1<br>MSAL1M_Alarm_Ack.ind(Rem_Add,<br>Alarm_Type:=Act_Alarm_PDU.Alarm_Type,<br>Slot_Number:=Act_Alarm_PDU.Slot_Number,<br>Seq_Nr:=Act_Alarm_PDU.Seq_Nr,<br>Alarm_Specifier:=Act_Alarm_PDU.Alarm_Specifier,<br>Add_Ack:=Act_Alarm_PDU.Add_Ack,<br>Alarm_Data:=Act_Alarm_PDU.Diagnostic_User_Data) | W-DIA-EVENT |
| 34 | CHK-DIA-ALARM | /Act_Alarm_PDU present &&<br>ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True &&<br>Alarm_Sequence = True && Alarm_Count < Alarm_Limit &&<br>Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type),<br>Act_Alarm_PDU.Seq_Nr] = w_con &&<br>Stored_Act_Alarm_PDU present<br>=>MSAL1M_Abort.ind(Rem_Add)<br>MSAL1M_Stopped.ind(Rem_Add) | WSTART |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 35 | CHK-DIA-ALARM | /Act_Alarm_PDU present && ALARM_ENABLED(Act_Alarm_PDU.Alarm_Type) = True && Alarm_Sequence = True && Alarm_Count < Alarm_Limit && Alarm_State_Table[Acls(Act_Alarm_PDU.Alarm_Type), Act_Alarm_PDU.Seq_Nr] = w_con && Stored_Alarm_PDU not present =>Stored_Alarm_PDU := Act_Alarm_PDU Alarm_Count := Alarm_Count+1 | W-DIA-EVENT |
| 36 | SEND-NEXT-ALARM | /Stored_Alarm_PDU not present =>ignore | READ-NEXT-ACK |
| 37 | SEND-NEXT-ALARM | /Stored_Alarm_PDU present =>Alarm_State_Table[Acls(Stored_Alarm_PDU.Alarm_Type), Stored_Alarm_PDU.Seq_Nr] := w_res Clear Stored_Alarm_PDU MSAL1M_Alarm_Notification.ind(Rem_Add, Alarm_Type:=Stored_Alarm_PDU.Alarm_Type, Slot_Number:=Stored_Alarm_PDU.Slot_Number, Seq_Nr:=Stored_Alarm_PDU.Seq_Nr, Alarm_Specifier:=Stored_Alarm_PDU.Alarm_Specifier, Add_Ack:=Stored_Alarm_PDU.Add_Ack, Alarm_Data:=Stored_Alarm_PDU.Diagnostic_User_Data) | READ-NEXT-ACK |
| 38 | READ-NEXT-ACK | /Alarm_Ack_FIFO_Filled = False =>ignore | W-DIA-EVENT |
| 39 | READ-NEXT-ACK | /Alarm_Ack_FIFO_Filled = True =>Load_From_Alarm_Ack_FIFO(New_Alarm_Type, New_Slot_Number, New_Seq_Nr) | SEND-NEXT-ACK |
| 40 | SEND-NEXT-ACK | /Alarm_State_Table[Acls(New_Alarm_Type), New_Seq_Nr] <> w_req =>MSAL1M_Abort.ind(Rem_Add) MSAL1M_Stopped.ind(Rem_Add) | WSTART |
| 41 | SEND-NEXT-ACK | /Alarm_State_Table[Acls(New_Alarm_Type), New_Seq_Nr] = w_req =>Alarm_State_Table[Acls(New_Alarm_Type), New_Seq_Nr] := w_con Alarm_Count := Alarm_Count-1 Waiting_For_Alarm_Ack := True MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number:=New_Slot_Number, Alarm_Type:=New_Alarm_Type, Seq_Nr:=New_Seq_Nr) | W-DIA-EVENT |
| 42 | ANY-STATE | MSAL1M_Reset.req(Rem_Add) =>MSAL1M_Reset.cnf(Rem_Add) | POWER-ON |

## 9.9 MSAC1M

### 9.9.1 Primitive definitions

#### 9.9.1.1 Primitives exchanged between FSPMM1 and MSAC1M

Table 83 shows the service primitives including their associated parameters issued by the FSPMM1 and received by the MSAC1M.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 83 – Primitives issued by FSPMM1 to MSAC1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Read.req | FSPMM1 | Rem Add,<br>Slot Number,<br>Index,<br>Length | |
| Write.req | FSPMM1 | Rem Add,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Alarm Ack.req | FSPMM1 | Rem Add,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |

Table 84 shows the service primitives including their associated parameters issued by the MSAC1M and received by the FSPMM1.

**Table 84 – Primitives issued by MSAC1M to FSPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Reject.ind | MSAC1M | Rem Add,<br>Status | |
| Read.cnf(+) | MSAC1M | Rem Add,<br>Length,<br>Data | |
| Read.cnf(-) | MSAC1M | Rem Add,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.cnf(+) | MSAC1M | Rem Add,<br>Length | |
| Write.cnf(-) | MSAC1M | Rem Add,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Alarm Ack.cnf(+) | MSAC1M | Rem Add,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Alarm Ack.cnf(-) | MSAC1M | Rem_Add,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

#### 9.9.1.2 Primitives exchanged between MSAC1M and MSAL1M

Table 85 shows the service primitives including their associated parameters issued by MSAL1M and received by the MSAC1M.

**Table 85 – Primitives issued by MSAL1M to MSAC1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MSAC1.req | MSAL1M | Rem Add | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | MSAL1M | Rem Add | |
| Start.req | MSAL1M | Rem Add | Start MS1-processing |
| Stop.req | MSAL1M | Rem Add | Stop MS1-processing |

Table 86 shows the service primitives including their associated parameters issued by MSAC1M and received by the MSAL1M.

**Table 86 – Primitives issued by MSAC1M to MSAL1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MSAC1.cnf | MSAC1M | Rem Add | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | MSAC1M | Rem Add | |
| Start.cnf | MSAC1M | Rem Add | Start completed |
| Stop.cnf | MSAC1M | Rem Add | Stop completed |
| Fault.ind | MSAC1M | (none) | |
| Abort.ind | MSAC1M | Rem Add | MSAC1 detected an error |

#### 9.9.1.3 Parameters of MSAL1M primitives

The parameters used with the primitives exchanged between the MSAL1M and MSCY1M are described in Table 87.

**Table 87 – Parameter used with primitives exchanged between MSAL1M and MSCY1M**

| Parameter name | Description |
|---|---|
| Rem Add | This parameter conveys the DL address of the assigned DP-slave. |

### 9.9.2 State machine description

The MSAC1M State Machine has the three states POWER-ON, CLOSED and OPEN. With the Init.req from the FSPMM1, the State Machine is initialised and changes to CLOSED. The state transition from CLOSED to OPEN is caused by the Start.req from the Slave-Handler (MSCY1M). The Slave-Handler is calling this service as soon as the DP-slave has entered the DATA-EXCH-state. When the DP-slave leaves the DATA-EXCH-state, the Stop.req service is called and the MSAC1M State Machine changes from OPEN to CLOSED. A Read/_Write request with an illegal parameter or a parallel service request while the last Read/Write is still being processed, is rejected to the User with a Reject.ind. An Alarm_Ack.req containing illegal service parameters or a parallel Alarm_Ack.req while the last Alarm_Ack.req is still being processed, is causing a state transition to POWER-ON and a Fault.ind to the FSPMM1.

The MSAC1M State Machine takes into account that the Alarm_Ack can be handled either on the SAP 51 as the Read/Write-service or on SAP 50.

The responses for Read and Write services are monitored by means of C1 Timer.

MSAC1M_Alarm_Ack is handled on the same SAP as MSAC1M_Read/Write.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

If a MSAC1M_Alarm_Ack is requested by the MSAL1M State Machine or a Read/_Write is requested by the User while no service is being processed, the service request is passed directly to the DL. If a Alarm_Ack.req is requested and a Read/Write is being processed, the Alarm_Ack.req is stored and sent as soon as the confirmation of the Read/Write is received. When a Read/Write.req is requested and a Alarm_Ack is being processed, the Read/Write is stored and sent as soon as the confirmation of the Alarm_Ack is received.

When recognising an error while a service request is being processed, the MSAC1M State Machine

— aborts the MS0 connection by indicating a Abort.ind to the MSCY1S,

— deletes an eventually stored service,

— informs the User with a Reject.ind in case of an outstanding Read/_Write service, and

— changes from state OPEN to CLOSE.

When a Stop.req is requested by the Slave-Handler, the MSAC1M first waits for an eventually outstanding DATA-REPLY.cnf. After this the Stop.req will be processed. A Read/_Write-confirmation is passed to the User and a possibly stored Alarm_Ack.req is deleted. In case of an Alarm_Ack confirmation, this confirmation is ignored, and a possibly outstanding Read/_Write is deleted and the User is informed with a Reject.ind. After that, the Stop.cnf is passed to the Slave-Handler and the MSAC1M State Machine changes from state OPEN to CLOSE.

MSAC1M_Alarm_Ack is handled on SAP 50

When a Alarm_Ack is requested by the MSAL1M State Machine or a Read/_Write service is requested by the User, the service request is passed directly to the DL.

When recognising an error while a service request is being processed on the other SAP, the MSAC1M State Machine

— waits for the confirmation of the service on the other SAP,

— then aborts the MSCY1M connection by indicating a Abort.ind to the Slave-Handler MSCY1S,

— informs the User with a Reject.ind in case of an outstanding _Read/_Write service, and

— changes from state OPEN to CLOSE.

When a Stop.req is requested by the Slave-Handler, the MSAC1M first waits for eventually outstanding DL-DATA-REPLY.cnf. After this the Stop.req will be processed. An outstanding MSAC1M_Read/_Write confirmation is passed to the User and an outstanding Alarm_Ack confirmation is ignored. Then the Stop.cnf is passed to the Slave-Handler and the MSAC1M State Machine changes from state OPEN to CLOSE.

**Local Variables**

**Go_Abort**
(Boolean)

This variable is set TRUE if the Stop.req is in use.

**Internal_Abort**
(Boolean)

This variable is set TRUE if an error in the Slave response occurs.

**Max_Data_Length**
(Unsigned8)

This variable defines the maximum size of MS1-PDU for all DP-slaves.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Alarm_DSAP**
(Unsigned8)

This variable contains the DSAP the Alarm_Ack is sent to.

**Service_Header**
(Unsigned8)

This variable contains the Function_Num of the MS1-Service actually sent to the DSAP 51.

**Service_Buffer**
(Octet-String)

This variable contains a Read or Write Request-PDU if a Alarm_Ack is in use (DSAP 51).

**Alarm_Buffer**
(Octet-String)

This variable contains a Alarm_Ack if a Read or Write Request-PDU is in use (DSAP 51).

**Src**
(Unsigned8)

This variable contains the number of the actual processed Read or Write Services (only 0 or 1 is possible).

**Arc**
(Unsigned8)

This variable contains the number of the actual Alarm_Ack Services (only 0 or 1 is possible)

**Stop_Pending**
(Boolean)

This variable is set TRUE if an Stop.cnf is pending.

**Start_C1_Monitoring**
(Boolean)

This variable is set TRUE if the user issues a Read or Write request service primitive. It is used to distinguish the first polling from later once and to start the C1 Timer for monitoring the user response.

**C1_Timer_Expired**
(Boolean)

This variable is set TRUE if the C1 Timer expires. With the next confirmation it is used to abort the connection if no response is available. It is only relevant if at least one polling for the response has been issued.

**C1_Timer**
Timer with 10 ms base

This timer is used to monitor the response of the DP-slave according the values specified in the GSD of the DP-slave (stored in the CRL). The function Start C1 Timer starts or restarts this timer with the value C1_Response_Timeout taken from the appropriate CRL entry. The function Stop C1 Timer stops the operation no matter of it was running before.

**Macros and Replacements**

**<REQPARAM>**

<

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

XXX=Read : Slot_Number, Index, Length
XXX=Write : Slot_Number, Index, Length, Data
>

**<VALID_SERVICE_FUNCTION_NUM>**

<
XXX=Read : 0x5E
XXX=Write : 0x5F
>

**<LOAD_SERVICE_BUFFER>**

<
XXX=Read : Service_Buffer[1]=0x5E
Service_Buffer[2]=Slot_Number
Service_Buffer[3]=Index
Service_Buffer[4]=Length
XXX=Write : Service_Buffer[1]=0x5F
Service_Buffer[2]=Slot_Number
Service_Buffer[3]=Index
Service_Buffer[4]=Length
Service_Buffer[5..Length+4]=Data
>

**<LOAD_ALARM_BUFFER>**

<
Alarm_Buffer[1]=0x5C
Alarm_Buffer[2]=Slot_Number
Alarm_Buffer[3]=Alarm_Type
Alarm_Buffer[4]=Seq_Nr*8
>

**IS_VALID_SERVICE_REQ**

(
Length ≤ Max_Data_Length
)

**IS_VALID_SERVICE_RES_PDU**

(
L_sdu[1] = Service_Buffer[1]
&& L_sdu.len ≥ 4
&& (L_sdu[1] <> 0x5E || L_sdu.len ≥ L_sdu[4]+4)
&& Max_Data_Length ≥ L_sdu[4]
)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**IS_VALID_SERVICE_NRS_PDU**

(
L_sdu[1] = (Service_Buffer[1]+0x80)
&& L_sdu.len = 4
)

**IS_ALARM_ACK_PDU**

(
L_sdu[2..4] = Alarm_Buffer[2..4]
&& (L_sdu[1] AND 0x7F = (Service_Buffer[1])
&& L_sdu.len = 4
)

**<VALID_SERVICE>**

<
Service_Header=0x5E : MSAC1M_Read
Service_Header=0x5F : MSAC1M_Write
>

**<VALID_SERVICE_RES_PARAM>**

<
Service_Header=0x5E : Length=L_sdu[4],
Data=L_sdu[5..Length+4]
Service_Header=0x5F : Length=L_sdu[4]
>

**<VALID_SERVICE_NRS_PARAM>**

<
Error_Decode=L_sdu[2],
Error_Code_1=L_sdu[3],
Error_Code_2=L_sdu[4],
>

**REJECT_VALID_SERVICE**

(
if (Stop_Pending=True)
MSAC1M_Reject.ind(Rem_Add, Status=REJ_SE)
else
if.(Go_Abort=True)
MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT)
else
if (Src <> 0)
MSAC1M_Reject.ind(Rem_Add, Status=REJ_PS)
else
if (Length<=Max_Data_Length)
MSAC1M_Reject.ind(Rem_Add, Status=REJ_LE)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

```
else
  MSAC1M_Reject.ind(Rem_Add, Status=REJ_IV)
)
```

**CONTINUE_VALID_SERVICE**

```
(
if (Src <> 0 && Alarm_Dsap=51)
  Service_Header = Service_Buffer[1]
  Start_C1_Monitoring = TRUE
  DMPMM1_DATA_REPLY.req(SSAP=51, DSAP=51,
   Rem_Add,L_sdu=Service_Buffer, Serv_class=Low)
)
```

**CONTINUE_ALARM_ACK**

```
(
if (Arc <> 0 && Alarm_Dsap=51)
  Service_Header = Alarm_Buffer[1]
  DMPMM1_DATA_REPLY.req(SSAP=51, DSAP=51, Rem_Add,
   L_sdu=Alarm_Buffer, Serv_class=Low)
)
```

**IS_INVALID_SERVICE_RES**

```
(
 (L_status = DL &&
 (L_sdu[1]<> Service_Header || L_sdu.len<4) &&
 (L_sdu[1]<> (Service_Header+0x80) || L_sdu.len<> 4)
 ) ||
 L_status = DS/UE/RS/RDL/NA/RR/LR/DH/RDH
)
```

**IS_INVALID_ALARM_ACK_RES**

```
(
 (L_status = DL &&
 (L_sdu[1..4]<> Alarm_Buffer[1..4] || L_sdu.len<> 4)
 ) ||
 L_status = DS/UE/RS/RDL/NA/RR/LR/DH/RDH
)
```

**CHECK_STOP_PENDING**

```
(
if (Stop_Pending = TRUE)
  MSAC1M_Stop.cnf(Rem_Add)
else
  MSAC1M_Abort.ind(Rem_Add)
)
```

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 9.9.3 MSAC1M state table

Table 88 contains the complete description of the MSAC1M state machine.

**Table 88 – MSAC1M state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | MSAC1M_MInit_MSAC1.req(Rem_Add)<br>=><br>MSAC1M_MInit_MSAC1.cnf(Rem_Add) | CLOSED |
| 2 | CLOSED | MSAC1M_Start.req(Rem_Add )<br>=><br>Alarm_Buffer := empty<br>Service_Buffer := empty<br>Stop_Pending := False<br>Go_Abort := False<br>Start_C1_Monitoring := FALSE<br>C1_Timer_Expired := FALSE<br>Src := 0<br>Arc := 0<br>Service_Header := 0<br>Max_Data_Length := Max_Channel_Data_Length<br>if (Extra_Alarm_Sap=FALSE) (Alarm_Dsap:=51) ELSE (Alarm_Dsap:=50)<br>MSAC1M_Start.cnf(Rem_Add) | OPEN |
| 3 | CLOSED | MSAC1M_Stop.req(Rem_Add)<br>=><br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |
| 4 | CLOSED | MSAC1M_XXX.req(Rem_Add, <REQPARAM>)<br>=><br>MSAC1M_Reject.ind(Rem_Add, Status:=REJ_SE) | CLOSED |
| 5 | CLOSED | MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>=><br>ignore | CLOSED |
| 6 | CLOSED | DMPMM1_DATA_REPLY.cnf(SSAP, DSAP, Rem_Add, L_sdu, Serv_class, L_status)<br>=><br>MSAC1M_Fault.ind(Rem_Add) | POWER-ON |
| 7 | OPEN | MSAC1M_Start.req(Rem_Add )<br>=><br>MSAC1M_Fault.ind(Rem_Add) | POWER-ON |
| 8 | OPEN | MSAC1M_Stop.req(Rem_Add)<br>/Arc = 0 && Src = 0<br>=><br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |
| 9 | OPEN | MSAC1M_Stop.req(Rem_Add)<br>/Arc <> 0 \|\| Src <> 0<br>=><br>Stop_Pending:=True | OPEN |
| 10 | OPEN | MSAC1M_XXX.req(Rem_Add, <REQPARAM>)<br>/Src = 0 && Stop_Pending = False && Go_Abort = False &&<br>IS_VALID_SERVICE_REQ && (Arc = 0 \|\| Alarm_Dsap <> 51)<br>=><br>Service_Header := <VALID_SERVICE_FUNCTION_NUM><br>Src := Src+1<br>Start_C1_Monitoring:=TRUE<br><LOAD_SERVICE_BUFFER><br>DMPMM1_DATA_REPLY.req(SSAP:=51, DSAP:=51, Rem_Add,<br>L_sdu:=Service_Buffer, Serv_class:=Low) | OPEN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 11 | OPEN | MSAC1M_XXX.req(Rem_Add, <REQPARAM>)<br>/Src = 0 && Stop_Pending = False && Go_Abort = False &&<br>IS_VALID_SERVICE_REQ && Arc <> 0 && Alarm_Dsap = 51<br>=><br>Src := Src+1<br><LOAD_SERVICE_BUFFER> | OPEN |
| 12 | OPEN | MSAC1M_XXX.req(Rem_Add, <REQPARAM>)<br>/Src <> 0 \|\| Stop_Pending = True \|\| Go_Abort = True \|\|<br>NOT (IS_VALID_SERVICE_REQ)<br>=><br>REJECT_VALID_SERVICE | OPEN |
| 13 | OPEN | MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/Arc = 0 && Stop_Pending = False && Go_Abort = False &&<br>Alarm_Dsap <> 51<br>=><br>Arc := Arc+1<br><LOAD_ALARM_BUFFER><br>DMPMM1_DATA_REPLY.req(SSAP:=51, DSAP:=Alarm_Dsap, Rem_Add,<br>L_sdu:=Alarm_Buffer, Serv_class:=Low)) | OPEN |
| 14 | OPEN | MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/Arc = 0 && Stop_Pending = False && Go_Abort = False &&<br>Src = 0 && Alarm_Dsap = 51<br>=><br>Arc := Arc+1<br>DMPMM1_DATA_REPLY.req(SSAP:=51, DSAP:=Alarm_Dsap, Rem_Add,<br>L_sdu:=Alarm_Buffer, Serv_class:=Low)) | OPEN |
| 15 | OPEN | MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/Arc = 0 && Stop_Pending = False && Go_Abort = False &&<br>Src <> 0 && Alarm_Dsap = 51<br>=><br>Arc := Arc+1<br><LOAD_ALARM_BUFFER> | OPEN |
| 16 | OPEN | MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/Arc=0 && (Stop_Pending = True \|\| Go_Abort = True)<br>=><br>ignore | OPEN |
| 17 | OPEN | MSAC1M_Alarm_Ack.req(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/Arc <> 0<br>=><br>DMPMM1_Fault.ind(Rem_Add, Status:=MSAC1_Fault) | POWER-ON |
| 18 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu,<br>Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_SERVICE_RES_PDU && Src <> 0 &&<br>L_sdu[1] = Service_Header && Stop_Pending = False &&<br>Go_Abort = False<br>=><br>Stop C1 Timer()<br>C1_Timer_Expired := FALSE<br>Src := 0<br>CONTINUE_ALARM_ACK<br>MSAC1_<VALID_SERVICE>.cnf(+)( Rem_Add, <VALID_SERVICE_RES_PARAM>) | OPEN |
| 19 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu,<br>Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_SERVICE_RES_PDU && Src <> 0 &&<br>L_sdu[1] = Service_Header && Stop_Pending = True &&<br>(Arc=0 \|\| Alarm_Dsap=51)<br>=><br>Stop C1 Timer()<br>C1_Timer_Expired := FALSE<br>Src := 0<br>Arc := 0<br>MSAC1_<VALID_SERVICE>.cnf(+)( Rem_Add, <VALID_SERVICE_RES_PARAM>)<br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 20 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_SERVICE_RES_PDU && Src <> 0 && L_sdu[1] = Service_Header && Stop_Pending = True && Arc <> 0 && Alarm_Dsap <> 51<br>=><br>Stop C1 Timer()<br>C1_Timer_Expired := FALSE<br>Src := 0<br>MSAC1_<VALID_SERVICE>.cnf(+)( Rem_Add, <VALID_SERVICE_RES_PARAM>) | OPEN |
| 21 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_SERVICE_RES_PDU && Src <> 0 && L_sdu[1] = Service_Header && Stop_Pending = False && Go_Abort = True<br>=><br>Stop C1 Timer()<br>C1_Timer_Expired := FALSE<br>Src := 0<br>MSAC1_<VALID_SERVICE>.cnf(+)( Rem_Add, <VALID_SERVICE_RES_PARAM>)<br>MSAC1M_Abort.ind(Rem_Add) | CLOSED |
| 22 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_SERVICE_NRS_PDU && Src <> 0 && L_sdu[1] = (Service_Header+0x80) && Stop_Pending = False && Go_Abort = False<br>=><br>Stop C1 Timer()<br>C1_Timer_Expired := FALSE<br>Src := 0<br>CONTINUE_ALARM_ACK<br>MSAC1_<VALID_SERVICE>.cnf(-)( Rem_Add, <VALID_SERVICE_NRS_PARAM>) | OPEN |
| 23 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_SERVICE_NRS_PDU && Src <> 0 && L_sdu[1] = (Service_Header+0x80) && Stop_Pending = True && (Arc=0 \|\| Alarm_Dsap=51)<br>=><br>Stop C1 Timer()<br>C1_Timer_Expired := FALSE<br>Src := 0<br>Arc := 0<br>MSAC1_<VALID_SERVICE>.cnf(-)( Rem_Add, <VALID_SERVICE_NRS_PARAM>)<br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |
| 24 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_SERVICE_NRS_PDU && Src <> 0 && L_sdu[1] = (Service_Header+0x80) && Stop_Pending = True && Arc <> 0 && Alarm_Dsap <> 51<br>=><br>Stop C1 Timer()<br>C1_Timer_Expired := FALSE<br>Src := 0<br>MSAC1_<VALID_SERVICE>.cnf(-)( Rem_Add, <VALID_SERVICE_NRS_PARAM>) | OPEN |
| 25 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_SERVICE_NRS_PDU && Src <> 0 && L_sdu[1] = (Service_Header+0x80) && Stop_Pending = False && Go_Abort = True<br>=><br>Stop C1 Timer()<br>C1_Timer_Expired := FALSE<br>Src := 0<br>MSAC1_<VALID_SERVICE>.cnf(-)( Rem_Add, <VALID_SERVICE_NRS_PARAM>)<br>MSAC1M_Abort.ind(Rem_Add) | CLOSED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 26 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_ALARM_ACK_PDU && Arc <> 0 && L_sdu[1]=Service_Buffer[1] && (Alarm_Dsap<>51 \|\| L_sdu[1]=Service_Header) && Stop_Pending = False && Go_Abort = False<br>=><br>Arc := 0<br>CONTINUE_VALID_SERVICE<br>MSAC1S_Alarm_Ack.cnf(+)( Rem_Add, Slot_Number, Alarm_Type, Seq_Nr) | OPEN |
| 27 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_ALARM_ACK_PDU && Arc <> 0 && L_sdu[1]=(Service_Buffer[1] OR 0x80) && (Alarm_Dsap<>51 \|\| L_sdu[1]=Service_Header) && Stop_Pending = False && Go_Abort = False<br>=><br>Arc := 0<br>CONTINUE_VALID_SERVICE<br>MSAC1S_Alarm_Ack.cnf(-)( Rem_Add, Slot_Number, Alarm_Type, Seq_Nr) | OPEN |
| 28 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_ALARM_ACK_RES_PDU && Arc <> 0 && (Alarm_Dsap<>51 \|\| L_sdu[1]=Service_Header) && Stop_Pending = True && Src = 0<br>=><br>Arc := 0<br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |
| 29 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_ALARM_ACK_PDU && Arc <> 0 && Alarm_Dsap = 51 && L_sdu[1] = Service_Header && Stop_Pending = True && Src <> 0<br>=><br>Arc := 0<br>Src := 0<br>MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT)<br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |
| 30 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_ALARM_ACK_PDU && Arc <> 0 && Alarm_Dsap <> 51 && Stop_Pending = True && Src <> 0<br>=><br>Arc := 0 | OPEN |
| 31 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = DL && IS_VALID_ALARM_ACK_PDU && Arc <> 0 && (Alarm_Dsap<>51 \|\| L_sdu[1]=Service_Header) && Stop_Pending = False && Go_Abort = True<br>=><br>Arc := 0<br>MSAC1M_Abort.ind(Rem_Add) | CLOSED |
| 32 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Src <> 0 && Service_Header <> Alarm_Ack && Stop_Pending = False && Go_Abort = False && Start_C1_Monitoring = TRUE<br>=><br>Start Timer C1()<br>Start_C1_Monitoring = FALSE<br>DMPMM1_DATA_REPLY.req(SSAP=51, DSAP, Rem_Add, L_sdu.len=0, Serv_class=Low) | OPEN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 33 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Src <> 0 && Service_Header <> Alarm_Ack && Stop_Pending = False && Go_Abort = False && Start_C1_Monitoring = FALSE && C1_Timer_Expired = FALSE<br>=><br>DMPMM1_DATA_REPLY.req(SSAP=51, DSAP, Rem_Add, L_sdu.len=0, Serv_class=Low) | OPEN |
| 34 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Src <> 0 && Service_Header <> Alarm_Ack && Stop_Pending = False && Go_Abort = False && Start_C1_Monitoring = FALSE && C1_Timer_Expired = TRUE<br>=><br>C1_Timer_Expired = FALSE<br>Src:=0<br>MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT)<br>MSAC1M_Abort.ind(Rem_Add) | CLOSED |
| 35 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Src <> 0 && Service_Header <> Alarm_Ack && Stop_Pending = True && (Arc=0 \|\| Alarm_Dsap=51)<br>=><br>Src := 0<br>Arc := 0<br>MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT)<br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |
| 36 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Src <> 0 && Service_Header <> Alarm_Ack && Stop_Pending = True && Arc <> 0 && Alarm_Dsap <> 51<br>=><br>Src := 0<br>MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT) | OPEN |
| 37 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Src <> 0 && Service_Header <> Alarm_Ack && Stop_Pending = False && Go_Abort = True<br>=><br>Src := 0<br>MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT)<br>MSAC1M_Abort.ind(Rem_Add) | CLOSED |
| 38 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Arc <> 0 && (Service_Header=Alarm_Ack \|\| Alarm_Dsap<>51) && Stop_Pending = False && Go_Abort = False<br>=><br>DMPMM1_DATA_REPLY.req(SSAP=51, DSAP, Rem_Add, L_sdu.len=0, Serv_class=Low) | OPEN |
| 39 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Arc <> 0 && (Service_Header=Alarm_Ack \|\| Alarm_Dsap<>51) && Stop_Pending = True && Src = 0<br>=><br>Arc := 0<br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |
| 40 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Arc <> 0 && Service_Header = Alarm_Ack && Alarm_Dsap = 51 && Stop_Pending = True && Src <> 0<br>=><br>Arc := 0<br>Src := 0<br>MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT)<br>MSAC1M_Stop.cnf(Rem_Add) | CLOSED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 41 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Arc <> 0 && Alarm_Dsap <> 51 &&<br>Stop_Pending = True && Src <> 0<br>=><br>Arc := 0 | OPEN |
| 42 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=Alarm_Dsap, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/L_status = NR && Arc <> 0 &&<br>(Service_Header=Alarm_Ack \|\| Alarm_Dsap<>51) &&<br>Stop_Pending = False && Go_Abort = True<br>=><br>Arc := 0<br>MSAC1M_Abort.ind(Rem_Add) | CLOSED |
| 43 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/IS_INVALID_SERVICE_RES && Src <> 0 &&<br>Service_Header <> Alarm_Ack && (Arc=0 \|\| Alarm_Dsap=51)<br>=><br>Src := 0<br>Arc := 0<br>CHECK_STOP_PENDING<br>MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT) | CLOSED |
| 44 | OPEN | DMPMM1_DATA_REPLY.cnf(SSAP=51, DSAP=51, Rem_Add, L_sdu, Serv_class=LOW, L_status)<br>/IS_INVALID_SERVICE_RES && Src <> 0 &&<br>Service_Header <> Alarm_Ack && Arc <> 0 && Alarm_Dsap <> 51<br>=><br>Src := 0<br>if Stop_Pending := False Go_Abort := True<br>MSAC1M_Reject.ind(Rem_Add, Status=REJ_ABORT) | OPEN |
| 45 | OPEN | C1 Timer expired<br>=><br>C1_Timer_Expired := TRUE | OPEN |
| 46 | CLOSED | C1 Timer expired<br>=><br>no action | CLOSED |

## 9.10 MMAC1

### 9.10.1 Primitive definitions

#### 9.10.1.1 Primitives exchanged between FSPMM1 and MMAC1

Table 89 shows the service primitives including their associated parameters issued by the FSPMM1 and received by the MMAC1.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 89 – Primitives issued by FSPMM1 to MMAC1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MM.req | FSPMM1 | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMM1 | (none) | |
| Abort.req | FSPMM1 | (none) | |
| Get Master Diag.rsp(+) | FSPMM1 | Diagnosis Data | |
| Get Master Diag.rsp(-) | FSPMM1 | Status | |
| Start Seq.rsp(+) | FSPMM1 | Max Len Data Unit | |
| Start Seq.rsp(-) | FSPMM1 | Status | |
| Download.rsp(+) | FSPMM1 | (none) | |
| Download.rsp(-) | FSPMM1 | Status | |
| Upload.rsp(+) | FSPMM1 | Data | |
| Upload.rsp(-) | FSPMM1 | Status | |
| End Seq.rsp(+) | FSPMM1 | (none) | |
| End Seq.rsp(-) | FSPMM1 | Status | |
| Act Param.rsp(+) | FSPMM1 | (none) | |
| Act Param.rsp(-) | FSPMM1 | Status | |

Table 90 shows the service primitives including their associated parameters issued by the MMAC1 and received by the FSPMM1.

**Table 90 – Primitives issued by MMAC1 to FSPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit MM.cnf | MMAC1 | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | MMAC1 | (none) | |
| Abort.ind | MMAC1 | (none) | |
| Fault.ind | MMAC1 | (none) | |
| Get Master Diag.ind | MMAC1 | MDiag Identifier | |
| Start Seq.ind | MMAC1 | Area Code,<br>Timeout | |
| Download.ind | MMAC1 | Area Code,<br>Add Offset,<br>Data | |
| Upload.ind | MMAC1 | Area Code,<br>Add Offset,<br>Data Len | |
| End Seq.ind | MMAC1 | (none) | |
| Act Param.ind | MMAC1 | Area Code,<br>Activate | |
| Act Para Brct.ind | MMAC1 | Area Code | |

### 9.10.1.2 Parameters of MMAC1 primitives

The parameters used with the primitives exchanged between the FSPMM1 and the MMAC1 are described in FAL Service Definition in IEC 61158-5-3.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

### 9.10.2 State machine description

The following MMAC1 state machine for Master-Master communication describes the responder side. Normally, this means a DP-interface of an automation device such as a Programmable Logic Controller.

After Reset, the MMAC1 changes to the “POWER-ON" state and waits for initialisation.

The state machine remains in the state “WAIT-REQ" as long as it receives a valid request from Layer 2. After issuing this request to the User, the state machine remains in the “WAIT-RES" state until the User offers the response. After storing the response, data collection is monitored. If a service-sequence is executed, MMAC1 is waiting for the next service request in the state “WAIT-SEQ-REQ".

At any time, maximally one request shall be in execution.

**Local Variables**

**T1**
(Unsigned16)

Time in which a Master (Class 2) shall receive at latest a response from the Master (Class 1).

**T2**
(Unsigned16)

Time that can elapse during the execution of a service-sequence between response and the following request.

**Client_Add**
(Unsigned8)

Address of the DP-master (Class 2) just communicating with.

**Service-Header**
(Octet-String of the length 4)

Intermediate memory for the first four octets of the last request.

**Sequence_Mode**
(Boolean)

Indicates that a service-sequence is just in execution. During a service-sequence the DP-master (Class 1) is reserved for the respective DP-master (Class 2).

### 9.10.3 MMAC1 state table

Table 91 contains the complete description of the MMAC1 state machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 91 – MMAC1 state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | MMAC1_Minit_MM.req<br>=>MMAC1_Minit_MM.cnf | WAIT-REQ |
| 2 | WAIT-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0<br>=>ignore | WAIT-REQ |
| 3 | WAIT-REQ | MMAC1_XXX.rsp<br>=>MMAC1_Reject.ind(Reason:=REJ_PS) | WAIT-REQ |
| 4 | WAIT-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=2 && L_sdu[1]=0x41<br>=>Service_Header:=L_sdu[1-2]<br>Client_Add:=Loc_add<br>MMAC1_Get_Master_Diag.ind(Identifier:=L_sdu[2])<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=Loc_add, Indication_Mode:=Unchanged) | WAIT-PROT |
| 5 | WAIT-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=4 && L_sdu[1]=0x42<br>=>Sequence_Mode:=True<br>Service_Header:=L_sdu[1-4]<br>T2:=L_sdu[3-4]<br>Client_Add:=Loc_add<br>MMAC1_Start_Seq.ind(Req_Add:=Loc_add, Area_Code:=L_sdu[2], Timeout:=L_sdu[3-4])<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=Loc_add, Indication_Mode:=Unchanged) | WAIT-PROT |
| 6 | WAIT-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len>=5 && L_sdu[1]=0x43<br>=>Service_Header:=L_sdu[1-4]<br>Client_Add:=Loc_add<br>MMAC1_Download.ind(Req_Add:=Loc_add, Area_Code:=L_sdu[2], Address_Offset:=L_sdu[3-4], Data:=L_sdu[5-L_sdu.len])<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=Loc_add, Indication_Mode:=Unchanged) | WAIT-PROT |
| 7 | WAIT-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=5 && L_sdu[1]=0x44<br>=>Service_Header:=L_sdu[1-4]<br>Client_Add:=Loc_add<br>MMAC1_Upload.ind(Req_Add:=Loc_add, Area_Code:=L_sdu[2], Address_Offset:=L_sdu[3-4], Data_Length:=L_sdu[5])<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=Loc_add, Indication_Mode:=Unchanged) | WAIT-PROT |
| 8 | WAIT-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=1 && L_sdu[1]=0x45<br>=>Client_Add:=Loc_add<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=Loc_add, Indication_Mode:=Unchanged) | END-SEQ-PROT |
| 9 | WAIT-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=3 && L_sdu[1]=0x47<br>=>Service_Header:=L_sdu[1-3]<br>Client_Add:=Loc_add<br>MMAC1_Act_Param.ind(Area_Code:=L_sdu[2], Activate:=L_sdu[3])<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=Loc_add, Indication_Mode:=Unchanged) | WAIT-PROT |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 10 | WAIT-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/invalid MMAC-REQ-PDU<br>=>Client_Add:=Loc_add<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=Loc_add, Indication_Mode:=Unchanged) | ERR-PROT |
| 11 | WAIT-REQ | DMPMM1_DATA.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=2 && L_sdu[1]=0x46<br>=>MMAC1_Act_Para_Brct.ind(Area_Code:=L_sdu[2]) | WAIT-REQ |
| 12 | WAIT-REQ | DMPMM1_DATA.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len<>2 \|\| L_sdu[1]<>0x46<br>=>ignore | WAIT-REQ |
| 13 | WAIT-SEQ-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0<br>=>ignore | WAIT-REQ |
| 14 | WAIT-SEQ-REQ | MMAC1_xxx.rsp<br>=>Sequence_Mode:=False<br>Stop T2<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=All, Indication_Mode:=Unchanged) | WAIT-UNPROT |
| 15 | WAIT-SEQ-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=2 && L_sdu[1]=0x41<br>=>Service_Header:=L_sdu[1-2]<br>Stop T2<br>MMAC1_Get_Master_Diag.ind(Identifier:=L_sdu[2]) | WAIT-RES |
| 16 | WAIT-SEQ-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=4 && L_sdu[1]=0x42<br>=>Sequence_Mode:=False<br>Stop T2<br>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu[1]:=0xC9, Serv_class:=Low, Transmit:=Single) {SE error code} | WAIT-UPD |
| 17 | WAIT-SEQ-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len>=5 && L_sdu[1]=0x43<br>=>Service_Header:=L_sdu[1-4]<br>Stop T2<br>MMAC1_Download.ind(Req_Add:=Loc_add, Area_Code:=L_sdu[2], Address_Offset:=L_sdu[3-4], Data:=L_sdu[5-L_sdu.len]) | WAIT-RES |
| 18 | WAIT-SEQ-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=5 && L_sdu[1]=0x44<br>=>Service_Header:=L_sdu[1-4]<br>Stop T2<br>MMAC1_Upload.ind(Req_Add:=Loc_add, Area_Code:=L_sdu[2], Address_Offset:=L_sdu[3-4], Data_Length:=L_sdu[5]) | WAIT-RES |
| 19 | WAIT-SEQ-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=1 && L_sdu[1]=0x45<br>=>Sequence_Mode:=False<br>Service_Header:=L_sdu[1]<br>Stop T2<br>MMAC1_End_Seq.ind(Req_Add:=Loc_add) | WAIT-RES |
| 20 | WAIT-SEQ-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=3 && L_sdu[1]=0x47<br>=>Service_Header:=L_sdu[1-3]<br>Stop T2<br>MMAC1_Act_Param.ind(Area_Code:=L_sdu[2], Activate:=L_sdu[3]) | WAIT-RES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 21 | WAIT-SEQ-REQ | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/invalid MMAC-REQ-PDU && Sequence_Mode=True<br>=>Sequence_Mode:=False<br>Stop T2<br>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu[1]:=0xC1, Serv_class:=Low, Transmit:=Single) {FE error code} | WAIT-UPD |
| 22 | WAIT-SEQ-REQ | T2 expired<br>=>Sequence_Mode:=False<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=All, Indication_Mode:=Unchanged) | WAIT-UNPROT |
| 23 | WAIT-SEQ-REQ | DMPMM1_DATA.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class)<br>/L_sdu.len=2 && Loc_add=Client_Add && L_sdu[1]=0x46<br>=>MMAC1_Act_Para_Brct.ind(Area_Code:=L_sdu[2]) | WAIT-SEQ-REQ |
| 24 | WAIT-SEQ-REQ | DMPMM1_DATA.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class)<br>/L_sdu.len<>2 \|\| Loc_add<>Client_Add \|\| L_sdu[1]<>0x46<br>=>ignore | WAIT-SEQ-REQ |
| 25 | END-SEQ-PROT | DMPMM1_RSAP_ACTIVATE.cnf(SSAP=54, M_status)<br>/M_status=OK<br>=>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu[1]:=0xC9, Serv_class:=Low, Transmit:=Single) {SE error code} | WAIT-UPD |
| 26 | ERR-PROT | DMPMM1_RSAP_ACTIVATE.cnf(SSAP=54, M_status)<br>/M_status=OK<br>=>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu[1]:=0xC1, Serv_class:=Low, Transmit:=Single) {FE error code} | WAIT-UPD |
| 27 | WAIT-PROT | DMPMM1_RSAP_ACTIVATE.cnf(SSAP=54, M_status)<br>/M_status=OK | WAIT-RES |
| 28 | WAIT-RES | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Sequence_Mode=False \|\| Client_Add<>Loc_add \|\| L_sdu.len=0<br>=>ignore | WAIT-RES |
| 29 | WAIT-RES | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Sequence_Mode=True && Client_Add=Loc_add && L_sdu.len>0<br>=>Sequence_Mode:=False<br>ignore L_sdu | WAIT-RES |
| 30 | WAIT-RES | MMAC1_Get_Master_Diag.rsp(Status, Diagnostic_Data)<br>/Service_Header[1]=0x41 && Status=OK<br>=>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu:=(Service_Header, Diagnostic_Data), Serv_class:=Low, Transmit:=Single) | WAIT-UPD |
| 31 | WAIT-RES | MMAC1_Start_Seq.rsp(Status, Max_Length_Data_Unit)<br>/Service_Header[1]=0x42 && Status=OK<br>=>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu:=(Service_Header, Max_Length_Data_Unit), Serv_class:=Low,Transmit:=Single) | WAIT-UPD |
| 32 | WAIT-RES | MMAC1_Download.rsp(Status)<br>/Service_Header[1]=0x43 && Status=OK<br>=>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu:=Service_Header, Serv_class:=Low,Transmit:=Single) | WAIT-UPD |
| 33 | WAIT-RES | MMAC1_Upload.rsp(Status, Data)<br>/Service_Header[1]=0x44 && Status=OK<br>=>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu:=(Service_Header, Data), Serv_class:=Low,Transmit:=Single) | WAIT-UPD |
| 34 | WAIT-RES | MMAC1_End_Seq.rsp(Status)<br>/Service_Header[1]=0x45 && Status=OK<br>=>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu:=Service_Header, Serv_class:=Low,Transmit:=Single) | WAIT-UPD |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 35 | WAIT-RES | MMAC1_Act_Param.rsp(Status)<br>/Service_Header[1]=0x47 && Status=OK<br>=>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu:=Service_Header,<br>Serv_class:=Low,Transmit:=Single) | WAIT-UPD |
| 36 | WAIT-RES | MMAC1_xxx.rsp(Status)<br>/Service_Header[1]=Service_Code && Status <> OK<br>=>Start T1<br>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu[1]:=Status,<br>Serv_class:=Low, Transmit:=Single) | WAIT-UPD |
| 37 | WAIT-RES | MMAC1_XXX.rsp(Status)<br>/Service_Header[1]<>Service_Code<br>=>Sequence_Mode:=False<br>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu[1]:=0xC6,<br>Serv_class:=Low, Transmit:=Single) {RE} | WAIT-UPD |
| 38 | WAIT-UPD | DMPMM1_REPLY_UPDATE.cnf(SSAP=54, Serv_class, L_status)<br>/Status=OK<br>=>Start T1 | WAIT-IND |
| 39 | WAIT-UPD | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class,<br>Update_status)<br>/Loc_add<>Client_Add && Update_status=NO<br>=>ignore | WAIT-UPD |
| 40 | WAIT-UPD | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class,<br>Update_status)<br>/Loc_add=Client_Add && L_sdu.len>0 && Update_status=NO<br>=>Sequence_Mode := False<br>ignore L_sdu | WAIT-UPD |
| 41 | WAIT-UPD | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class,<br>Update_status)<br>/Update_status=LO/HI<br>=>MMAC1_Fault.ind | POWER-ON |
| 42 | WAIT-UPD | DMPMM1_REPLY_UPDATE.cnf(SSAP=54)<br>/Status=NO/IV<br>=>MMAC1_Fault.ind | POWER-ON |
| 43 | WAIT-UPD | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class,<br>Update_status)<br>/(Update_status=LO/HI) \|\| (Update_status=NO && L_sdu.len=0)<br>=>MMAC1_Fault.ind | POWER-ON |
| 44 | WAIT-UPD | DMPMM1_DATA.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class)<br>/L_sdu.len=2 && L_sdu[1]=0x46<br>&& (Sequence_Mode=False \|\| Loc_add=Client_Add)<br>=>MMAC1_Act_Para_Brct.ind(Area_Code:=L_sdu[2]) | WAIT-UPD |
| 45 | WAIT-UPD | DMPMM1_DATA.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class)<br>/L_sdu.len<>2 \|\| L_sdu[1]<>0x46 \|\| Sequence_Mode=True \|\| Loc_add<>Client_Add<br>=>ignore | WAIT-UPD |
| 46 | WAIT-IND | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class,<br>Update_status)<br>/Loc_add<>Client_Add && Update_status=NO<br>=>ignore | WAIT-IND |
| 47 | WAIT-IND | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class,<br>Update_status)<br>/Update_status=LO/HI && Loc_add<>Client_Add<br>=>stop T1<br>MMAC1_Fault.ind | POWER-ON |
| 48 | WAIT-IND | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class,<br>Update_status)<br>/(Update_status=LO && Loc_add<>Client_Add) \|\|(Update_status=NO && L_sdu.len=0)<br>\|\| (Update_status=HI)<br>=>MMAC1_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 49 | WAIT-IND | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Loc_add=Client_Add && L_sdu.len>0 && Update_status=NO<br>=>Sequence_Mode = False<br>ignore L_sdu | WAIT-IND |
| 50 | WAIT-IND | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && Sequence_Mode=False && Loc_add = Client_Add<br>=>ignore L_sdu, if existent<br>Stop T1<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=All, Indication_Mode:=Unchanged) | WAIT-UNPROT |
| 51 | WAIT-IND | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && Sequence_Mode=True && L_sdu.len=0<br>&& Loc_add = Client_Add<br>=>Stop T1<br>Start T2 | WAIT-SEQ-REQ |
| 52 | WAIT-IND | DMPMM1_DATA_REPLY.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && Sequence_Mode=True && L_sdu.len>0<br>&& Loc_add = Client_Add<br>=>Sequence_Mode:=False<br>ignore L_sdu, if existent<br>STOP T1<br>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=All, Indication_Mode:=Unchanged) | WAIT-UNPROT |
| 53 | WAIT-IND | DMPMM1_DATA.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class)<br>/L_sdu.len=2 && L_sdu[1]=0x46<br>&& (Sequence_Mode=False \|\| Loc_add=Client_Add)<br>=>MMAC1_Act_Para_Brct.ind(Area_Code:=L_sdu[2]) | WAIT-IND |
| 54 | WAIT-IND | DMPMM1_DATA.ind(SSAP,DSAP=54,Loc_add,Rem_add,L_sdu,Serv_class)<br>/L_sdu.len<>2 \|\| L_sdu[1]<>0x46 \|\| (Sequence_Mode=True && Loc_add<>Client_Add)<br>=>ignore | WAIT-IND |
| 55 | WAIT-IND | T1 expired<br>=>Sequence_Mode:=False<br>DMPMM1_REPLY_UPDATE.req(SSAP:=54, L_sdu.len:=0,<br>Serv_class:=Low, Transmit:=Single) | WAIT-DEL |
| 56 | WAIT-DEL | DMPMM1_REPLY_UPDATE.cnf(SSAP=54)<br>/L_status=OK<br>=>DMPMM1_RSAP_ACTIVATE.req(SSAP:=54, Access:=All, Indication_Mode:=Unchanged) | WAIT-UNPROT |
| 57 | ANY-STATE | MMAC1_Reset.req<br>=>Stop T1<br>MMAC1_Reset.cnf | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

## 9.11 MSCS1M

### 9.11.1 Primitive definitions

#### 9.11.1.1 Primitives exchanged between MSCS1M and FSPMM1

Table 92 shows the service primitives including their associated parameters issued by the FSPMM1 and received by the MSCS1M.

**Table 92 – Primitives issued by FSPMM1 to MSCS1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Set Time.req | FSPMM1 | AREP,<br>Time Value,<br>Local Time Diff,<br>Summertime,<br>Accuracy,<br>Synchronisation Active,<br>Announcement Hour | Refer to FAL Service Definition in IEC 61158-5-3 |

Table 93 shows the service primitives including their associated parameters issued by the MSCS1M and received by the FSPMM1.

**Table 93 – Primitives issued by MSCS1M to FSPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Set Time.ind | MSCS1M | AREP,<br>Time Value,<br>Local Time Diff,<br>Summertime,<br>Accuracy,<br>Synchronisation Active,<br>Announcement Hour | Refer to FAL Service Definition in IEC 61158-5-3 |
| Set Time.cnf | MSCS1M | AREP,<br>Status | |
| Sync Interval Violation.ind | MSCS1M | AREP | |

#### 9.11.1.2 Parameter of MSCS1M primitives

The parameters used with the primitives exchanged between the FSPMM1 and the MSCS1M are described in FAL Service Definition in IEC 61158-5-3.

### 9.11.2 State machine description

In IDLE state this machine waits either for a Clock Value indication from a valid Time master conveyed by DMPMM1 to enter StopTM state and deactivate the own Time master capability or for a Set Time request from the local user conveyed by the FSPMM1 to enter WTEc state and become an active Time master.

In the states WTEc and WCVc the machine issues a clock synchronisation sequence. On completion of this sequence a Set Time confirmation is issued and the machine returns to IDLE. In case of an interruption of the sequence by a valid Clock Value indication the machine brances to StopTM state and the Time master capability becomes inactive.

The StopTM state will be left, if no valid clock synchronisation sequence is detected.

**Local variables of the MSCS1M**

**Time Last Rcvd**
(Network Time)

This local variable contains the Clock_Value_Time_Event of the last valid received Clock Value indication.

**Error**
(Unsigned8)

This local variable is a counter for invalid Clock Synchronisation sequences. It will be reset to 0, when a Sync Interval Violation is indicated.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Clock**
(Network Time)

This Variable is used to hold the time value between issuing a Time Event service request and receiving the corresponding Time Event service confirmation.

**T_State**
(Compound)

This local variable is used for internal storage of service parameters for clock synchronisation.

**T_Last**
(Network Time)

This local variable is used for internal storage of the Time Value of the last valid clock synchronisation sequence. It is used to check the validity of subsequent clock synchronisations.

**Macros of the MSCS1M:**

**SET_TIME_ATTRIBUTES**

(
Local Time Diff :=CS_list.Clock_value_status.C * (-1)^CS_list.Clock_value_status.CV
Summertime := CS_list.Clock_value_status.SWT
Accruacy :=CS_list.Clock_value_status.CR
Synchronisation Active := CS_list.Clock_value_status.SYF
Announcement Hour :=CS_list.Clock_value_status.ANH
)

**GET_TIME_ATTRIBUTES**

(
CS_list.Clock_value_status.C * (-1)^CS_list.Clock_value_status.CV :=Local Time Diff
CS_list.Clock_value_status.SWT := Summertime
CS_list.Clock_value_status.CR := Accruacy
CS_list.Clock_value_status.SYF := Synchronisation Active
CS_list.Clock_value_status.ANH := Announcement Hour
)

### 9.11.3 MSCS1M state table

Table 94 contains the complete description of the MSCS1M state machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 94 – MSCS1M state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == SV && Error < 2<br>=><br>Error := Error+1 | Idle |
| 2 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE != Time Last Rcvd && Error < 2<br>=><br>Time Last Rcvd:= CS_list.Clock_Value_Time_Event<br>Error := Error+1 | Idle |
| 3 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == SV && Error >= 2<br>=><br>Error := 0<br>Sync Interval Violation.ind | Idle |
| 4 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE != Time Last Rcvd && Error >= 2<br>=><br>Time Last Rcvd:= CS_list.Clock_Value_Time_Event<br>Error := 0<br>Sync Interval Violation.ind | Idle |
| 5 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE == Time Last Rcvd &&<br>!(Time_Master_Addr < TS)<br>=><br>ignore | Idle |
| 6 | Idle | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE == Time Last Rcvd &&<br>Time_Master_Addr < TS<br>=><br>Error := 0<br>Time Last Rcvd:= CS_list.Clock_Value_Time_Event<br>Time Value := Time Last Rcvd + Receive Delay Time<br>Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour from CS-list.Clock_Value_Status<br>Set_Time.ind(Time Value, Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour) | StopTM |
| 7 | Idle | Set_Time.req(Time Value, Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour)<br>=><br>T_State from Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour<br>Clock:=Time Value<br>CS-TIME_EVENT.req() | WTEc |
| 8 | WTEc | CS-TIME-EVENT.cnf ( Send_Delay_Time, DL_status)<br>/DL_status = OK<br>=><br>CS_list.Clock_Value_Previous_TE:= T_Last<br>T_Last:= Clock + Send_Delay_Time<br>CS_list.Clock_Value_Time_Event:= T_Last<br>DL-CS-CLOCK-VALUE.req (CS_list ) | WCVc |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 9 | WTEc | CS-TIME-EVENT.cnf ( Send_Delay_Time, DL_status)<br>/DL_status != OK<br>=><br>Status:= DL_Status<br>Set_Time.cnf(Status) | Idle |
| 10 | WTEc | Set_Time.req(Time Value, Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour)<br>=><br>Status:= SV<br>Set_Time.cnf(Status) | WTEc |
| 11 | WCVc | CS-CLOCK-VALUE.cnf (DL_status)<br>=><br>Status:= DL_Status<br>Set_Time.cnf(Status) | Idle |
| 12 | WCVc | Set_Time.req(Time Value, Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour)<br>=><br>Status:= SV<br>Set_Time.cnf(Status) | WCVc |
| 13 | StopTM | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == SV && Error < 2<br>=><br>Error := Error+1 | StopTM |
| 14 | StopTM | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE != Time Last Rcvd && Error < 2<br>=><br>Time Last Rcvd:= CS_list.Clock_Value_Time_Event<br>Error := Error+1 | StopTM |
| 15 | StopTM | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == SV && Error >= 2<br>=><br>Error := 0<br>Sync Interval Violation.ind | Idle |
| 16 | StopTM | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE != Time Last Rcvd && Error >= 2<br>=><br>Time Last Rcvd:= CS_list.Clock_Value_Time_Event<br>Error := 0<br>Sync Interval Violation.ind | Idle |
| 17 | StopTM | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE == Time Last Rcvd &&<br>Time_Master_Addr < TS<br>=><br>Error := 0<br>Time Last Rcvd:= CS_list.Clock_Value_Time_Event<br>Time Value := Time Last Rcvd + Receive Delay Time<br>Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour from CS-list.Clock_Value_Status<br>Set_Time.ind(Time Value, Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour) | StopTM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 18 | StopTM | CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time)<br>/CS_status == OK &&<br>CS_list.Clock_Value_previous_TE == Time Last Rcvd &&<br>!(Time_Master_Addr < TS)<br>=><br>ignore | StopTM |
| 19 | StopTM | Set_Time.req(Time Value, Local Time Diff, Summertime, Accuracy, Synchronisation Active, Announcment Hour)<br>=><br>Status:= SV<br>Set_Time.cnf(Status) | StopTM |

## 9.12 MSAC2M

### 9.12.1 Primitive definitions

#### 9.12.1.1 Primitives exchanged between FSPMM2 and MSAC2M

Table 95 shows the service primitives including their associated parameters issued by FSPMM2 and received by the MSAC2M.

**Table 95 – Primitives issued by FSPMM2 to MSAC2M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit MS2.req | FSPMM2 | C_Ref | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMM2 | (none) | |
| Abort.req | FSPMM2 | C_Ref | |
| Initiate.req | FSPMM2 | C_Ref,<br>Send Timeout,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Read.req | FSPMM2 | C_Ref,<br>Slot Number,<br>Index,<br>Length | |
| Write.req | FSPMM2 | C_Ref,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Data Transport.req | FSPMM2 | C_Ref,<br>Slot Number,<br>Index,<br>Length,<br>Data | |

Table 96 shows the service primitives including their associated parameters issued by the MSAC2M and received by the FSPMM2.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 96 – Primitives issued by MSAC2M to FSPMM2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit MS2.cnf | MSAC2M | C_Ref | |
| Reset.cnf | MSAC2M | (none) | |
| Initiate.cnf(+) | MSAC2M | C_Ref,<br>Max Len Data Unit,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Initiate.cnf(-) | MSAC2M | C_Ref,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Read.cnf(+) | MSAC2M | C_Ref,<br>Length,<br>Data | |
| Read.cnf(-) | MSAC2M | C_Ref,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.cnf(+) | MSAC2M | C_Ref,<br>Length | |
| Write.cnf(-) | MSAC2M | C_Ref,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Data Transport.cnf(+) | MSAC2M | C_Ref,<br>Length,<br>Data | |
| Data Transport.cnf(-) | MSAC2M | C_Ref,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Reject.ind | MSAC2M | C_Ref,<br>Reason Code | |
| Abort.ind | MSAC2M | C_Ref,<br>Locally_Generated,<br>Subnet,<br>Instance,<br>Reason_Code,<br>Abort_Detail | |
| Fault.ind | MSAC2M | C_Ref | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 9.12.1.2 Parameters of MSAC2M primitives

The parameters used with the primitives exchanged between the FSPMM2 and the MSAC2M are described in FAL Service Definition in IEC 61158-5-3. Additional parameters are described in Table 97.

**Table 97 – Parameters used with primitives exchanged with MSAC2M**

| Parameter name | Description |
|---|---|
| C_Ref | Local identifier of a MSAC2 state machine |

### 9.12.2 State machine description

On POWER-ON each MSAC2M State Machine waits for initialisation.

In the state CLOSED the Initiate service called from the AP-Context is expected and sent to the DP-slave. In the state START-POLL-RES the Master is waiting for the positive Initiate.cnf and makes the transition into the state OPEN.

In the OPEN-state the services Read, Write and Data_Transport are allowed. The State Machine controls that only one request from the User is processed simultaneously, otherwise the connection is aborted.

The connection can be aborted by the User of the Master or by the User of the Slave.It is aborted automatically if a protocol error is detected.

**Local Variables**

**Client_SAP**
(Unsigned8)

The SAP used for the connection.

**Server_SAP**
(Unsigned8)

The SAP used for the connection at the server side.

**Server_Add**
(Unsigned8)

The DL address (0-126) of the remote server.

**S-Timer**
(Unsigned16)

This timer monitors the sending of service request PDUs on the connection if no service is outstanding. The S-Timer expires if no service request PDU has been sent during the S-Timer interval. In this case an Idle-REQ-PDU will be sent.

**R-Timer**
(Unsigned16)

This Timer monitors the liveliness of the connection if there is an outstanding service response. The R-Timer expires if no service response or IDLE-PDU during the R-Timer interval has been received. In this case the connection will be aborted. The R-Timer interval is 3 times the S-Timer interval.

**Service-Header**
Local buffer for the Function_Num of the last transmitted request.

**Go_Abort**
(Boolean)

Local flag that indicates an abort reaction after receiving the remote response.

**Service_Buffer**
(Octet-String[255])

Local buffer that stores one complete service during an outstanding Idle-RES-PDU.

**Macros**
;XXX handles Data_Transport, Read and Write Services.

**<REQPARAM>**

<

XXX=Read : Slot_Number, Index, Length

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

XXX=Write :Slot_Number, Index, Length, Data
XXX=Data_Transport : Slot_Number, Index, Length, Data
>

**<VALID_SERVICE_FUNCTION_NUM>**
<
XXX=Read : 0x5E
XXX=Write : 0x5F
XXX=Data_Transport : 0x51
>

**<LOAD_SERVICE_BUFFER>**
<
XXX=Read : Service_Buffer[1]=0x5E
Service_Buffer[2]=Slot_Number
Service_Buffer[3]=Index
Service_Buffer[4]=Length
XXX=Write : Service_Buffer[1]=0x5F
Service_Buffer[2]=Slot_Number
Service_Buffer[3]=Index
Service_Buffer[4]=Length
Service_Buffer[5..Length+4]=Data
XXX=Data_Transport : Service_Buffer[1]=0x51
Service_Buffer[2]=Slot_Number
Service_Buffer[3]=Index
Service_Buffer[4]=Length
Service_Buffer[5..Length+4]=Data
>

**VALID_SERVICE**
(
Length ≤ Max_Data_Length
)

**VALID_SERVICE_RES_PDU**
(
L_sdu[1] = Service_Buffer[1] &&
L_sdu.len ≥ 4 &&
(L_sdu[1] <> 0x5E || L_sdu.len ≥ L_sdu[4]+4) &&
Max_Data_Length ≥ L_sdu[4]
)

**VALID_SERVICE_NRS_PDU**
(

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

```
 L_sdu[1] = (Service_Buffer[1]+0x80) &&
 L_sdu.len = 4
)

<VALID_SERVICE>
<
 Service_Header=0x5E : Read
 Service_Header=0x5F : Write
 Service_Header=0x51 : Data_Transport
>

<VALID_SERVICE_RES_PARAM>
<
Service_Header=0x5E : Length=L_sdu[4],
   Data=L_sdu[5..Length+4]
Service_Header=0x5F : Length=L_sdu[4]
Service_Header=0x51 : Length=L_sdu[4],
   Data=L_sdu[5..Length+4]
>

<VALID_SERVICE_NRS_PARAM>
<
Error_Decode=L_sdu[2],
Error_Code_1=L_sdu[3],
Error_Code_2=L_sdu[4],
>

VALID_SERVICE_REQ_PDU
(
 (Read-REQ-PDU) ||
 (Write-REQ-PDU) ||
 (Data_Transport-REQ-PDU)
)

VALID_SERVICE_RES_PDU
(
 (Read-RES-PDU) ||
 (Write-RES-PDU) ||
 (Data_Transport-RES-PDU)
)

VALID_SERVICE_NRS_PDU
(
 (Read-NRS-PDU) ||
 (Write-NRS-PDU) ||
```

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

(Data_Transport-NRS-PDU)
)

**STORE_ABORT_PARAMETER**
(
Stored-Subnet=Subnet
Stored-Instance=Instance
Stored-Reason_Code=Reason_Code
)

### 9.12.3 MSAC2M state table

Table 98 contains the complete description of the MSAC2M state machine.

**Table 98 – MSAC2M state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | MSAC2M_Minit_MS2.req(C_Ref)<br>=><br>Stored_RM_SAP := 49<br>Stored_C_Ref := C_Ref<br>Stored_Max_Local_Data_Length := Max_Local_Data_Length<br>Client_SAP := 50<br>MSAC2M_Minit_MS2.cnf(C_Ref) | CLOSED |
| 2 | CLOSED | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>Server_Add := Rem_Add<br>S-Timer := Send_Timeout<br>R-Timer := 3*Send_Timeout<br>Go_Abort := FALSE<br>Service_Buffer := empty<br>store Initiate-REQ-PDU<br>Start S-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=STORED_RM_SAP, Rem_Add:=Server_Add, L_sdu:=Initiate-REQ-PDU, Serv_class:=Low) | START-POLL-RMREQ |
| 3 | CLOSED | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 4 | CLOSED | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 5 | CLOSED | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_RM_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = IV/LS<br>=><br>ignore | CLOSED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 6 | START-POLL-RMREQ | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_RM_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = RM-REQ-PDU &&<br>S-Timer >= Send_Timeout && Go_Abort = FALSE<br>=><br>Stored_Server_SAP := Server_SAP<br>Stop S-Timer<br>Start R-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu.len:=0, Serv_class:=Low) | START-POLL-RES |
| 7 | START-POLL-RMREQ | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_RM_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = RM-REQ-PDU && Go_Abort = TRUE<br>=><br>Stored_Server_SAP := Server_SAP<br>Stop S-Timer<br>Start R-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu.len:=0, Serv_class:=Low) | START-POLL-RES |
| 8 | START-POLL-RMREQ | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_RM_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = RM-REQ-PDU && S-Timer < Send_Timeout<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_STO,<br>Additional_Detail:=Send_Timeout,<br>STORE_ABORT_PARAMETER<br>Stored_Server_SAP := Server_SAP<br>Go_Abort := TRUE<br>Stop S-Timer<br>Start R-Timer<br>MSAC2M_Abort.ind (C_Ref, Locally_Generated,<br>Subnet, Instance, Reason_Code, Additional_Detail)<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu.len:=0, Serv_class:=Low) | START-POLL-RES |
| 9 | START-POLL-RMREQ | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_RM_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu <> RM-REQ-PDU<br>=><br>Stop S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_FE)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 10 | START-POLL-RMREQ | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_RM_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = RR && Go_Abort = FALSE<br>=><br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=STORED_RM_SAP, Rem_Add:=Server_Add, L_sdu:=stored Initiate-REQ-PDU, Serv_class:=Low) | START-POLL-RMREQ |
| 11 | START-POLL-RMREQ | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_RM_SAP, Rem_add=Server_Add, L_sdu,Serv_class=Low, L_Status)<br>/L_status = RR && Go_Abort = TRUE<br>=><br>Stop S-Timer<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 12 | START-POLL-RMREQ | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_RM_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = RS/NR/RDL/RDH/LR/NA/DH/DS/UE<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=FALSE, Instance:=DLL, Reason_Code:=L_status<br>ignore L_sdu if present<br>Stop S-Timer<br>MSAC2M_Abort.ind (C_Ref, Locally_Generated,<br>Subnet, Instance, Reason_Code)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 13 | START-POLL-RMREQ | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER | START-POLL-RMREQ |
| 14 | START-POLL-RMREQ | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE,<br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated,<br>Subnet, Instance, Reason_Code) | START-POLL-RMREQ |
| 15 | START-POLL-RMREQ | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE,<br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated,<br>Subnet, Instance, Reason_Code) | START-POLL-RMREQ |
| 16 | START-POLL-RMREQ | S-Timer expired<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_TO) | FINAL-WAIT-CON |
| 17 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP,<br>Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = RS/NR<br>=><br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP,<br>Rem_Add:=Server_Add, L_sdu.len:=0, Serv_class:=Low) | START-POLL-RES |
| 18 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP,<br>Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Initiate-RES-PDU && Go_Abort = FALSE<br>=><br>Stored_Max_Data_Lenth := Max_Len_Data_Unit<br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_Initiate.cnf(+, C_Ref:=Stored_C_Ref,<br>Features_Supported,<br>Profile_Features_Supported, Profile_Ident_Number,<br>Add_Addr_Param) | OPEN |
| 19 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP,<br>Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Initiate-RES-PDU && Go_Abort = TRUE<br>=><br>Stop R-Timer<br>Start S-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP,<br>Rem_Add:=Server_Add,<br>L_sdu:=Abort-REQ-PDU with stored Parameter,<br>Serv_class:=Low) | END-POLL-RES |
| 20 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP,<br>Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DS/UE<br>=><br>Stop R-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=DLL, Reason_Code:=L_status)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 21 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = RDL/NA/LR/DH/RDH/RR<br>=><br>ignore L_sdu if present<br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=DLL, Reason_Code:= L_status)<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU, Serv_class:=Low) | END-POLL-RES |
| 22 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Abort-REQ-PDU<br>=><br>Stop R-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=FALSE, Subnet, Instance, Reason_Code)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 23 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = (Initiate-NRS-PDU) && Go_Abort = FALSE<br>=><br>Stop R-Timer<br>MSAC2M_Initiate.cnf(-,C_Ref:=Stored_C_Ref, Error_Decode,Error_Code_1,Error_Code_2)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 24 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = (Initiate-NRS-PDU) && Go_Abort = TRUE<br>=><br>Stop R-Timer<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 25 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL &&<br>NOT (L_sdu = Initiate-RES-PDU \|\| L_sdu = (Initiate-NRS-PDU) \|\| L_sdu = Abort-REQ-PDU \|\| L_sdu = Idle-REQ-PDU)<br>=><br>ignore L_sdu<br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_RE)<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU, Serv_class:=Low) | END-POLL-RES |
| 26 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Idle-REQ-PDU && Go_Abort = FALSE<br>=><br>Stop R-Timer<br>Start R-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Idle-RES-PDU, Serv_class:=Low) | START-POLL-RES |
| 27 | START-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Idle-REQ-PDU && Go_Abort = TRUE<br>=><br>Stop R-Timer<br>Start S-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU with stored Parameter, Serv_class:=Low) | END-POLL-RES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 28 | START-POLL-RES | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE,<br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated,<br>Subnet, Instance, Reason_Code) | START-POLL-RES |
| 29 | START-POLL-RES | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER | START-POLL-RES |
| 30 | START-POLL-RES | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE,<br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated,<br>Subnet, Instance, Reason_Code) | START-POLL-RES |
| 31 | START-POLL-RES | R-Timer expired<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_TO<br>STORE_ABORT_PARAMETER<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated,<br>Subnet, Instance, Reason_Code) | END-WAIT-CON |
| 32 | OPEN | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>/Length <= (Stored_Max_Data_Length) &&<br>Length <= (Stored_Max_Local_Data_Length)<br>=><br>Service_Header := Function_Num<br>Stop S-Timer<br>Start R-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP,<br>Rem_Add:=Server_Add, L_sdu:=VALID_SERVICE_REQ_PDU,<br>Serv_class:=Low) | VALID-SERVICE-POLL-RES |
| 33 | OPEN | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>/Length > (Stored_Max_Data_Length) ||<br>Length > (Stored_Max_Local_Data_Length)<br>=><br>MSAC2M_Reject.ind(C_Ref:=Stored_C_Ref, Reason_Code:=REJ_LE) | OPEN |
| 34 | OPEN | S-Timer expired<br>=><br>Start R-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP,<br>Rem_Add:=Server_Add, L-sdu:=Idle-REQ-PDU, Serv_class:=Low) | IDLE-POLL-RES |
| 35 | OPEN | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>Stop S-Timer<br>Start S-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP,<br>Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU, Serv_class:=Low) | END-POLL-RES |
| 36 | OPEN | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>Stop S-Timer<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE)<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP,<br>Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU, Serv_class:=Low) | END-POLL-RES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 37 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = NR<br>=><br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu.len:=0, Serv_class:=Low) | VALID-SERVICE-POLL-RES |
| 38 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Idle-REQ-PDU<br>=><br>Stop R-Timer<br>Start R-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Idle-RES-PDU, Serv_class:=Low) | VALID-SERVICE-POLL-RES |
| 39 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = VALID_SERVICE_RES_PDU &&<br>Function_Num = Service_Header && Go_Abort = FALSE<br>=><br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_<VALID_SERVICE>.cnf(+, C_Ref:=Stored_C_Ref, <VALID_SERVICE_RES_PARAM>) | OPEN |
| 40 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = VALID_SERVICE_NRS_PDU &&<br>Function_Num = (Service_Header+0x80) && Go_Abort = FALSE<br>=><br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_<VALID_SERVICE>.cnf(-, C_Ref:=Stored_C_Ref, Rem_Add, <VALID_SERVICE_NRS_PARAM>) | OPEN |
| 41 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && NOT(Abort-REQ-PDU) && Go_Abort = TRUE<br>=><br>ignore L_sdu<br>Stop R-Timer<br>Start S-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add,<br>L_sdu:=Abort-REQ-PDU with stored Parameter,<br>Serv_class:=Low) | END-POLL-RES |
| 42 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DS/UE/RS<br>=><br>Stop R-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=DLL, Reason_Code:=L_status)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 43 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && Go_Abort = FALSE &&<br>NOT(L_sdu = VALID_SERVICE_RES_PDU &&<br>(Function_Num = Service_Header)) &&<br>NOT(L_sdu = VALID_SERVICE_NRS_PDU &&<br>(Function_Num = (Service_Header+0x80))) &&<br>NOT(L_sdu = Abort-REQ-PDU) &&<br>NOT(L_sdu = Idle-REQ-PDU)<br>=><br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_RE)<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU, Serv_class:=Low) | END-POLL-RES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 44 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = RDL/NA/RR/LR/DH/RDH<br>=><br>ignore L_sdu if present<br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=DLL, Reason_Code:=L_status)<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU, Serv_class:=Low) | END-POLL-RES |
| 45 | VALID-SERVICE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Abort-REQ-PDU<br>=><br>Stop R-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=FALSE, Subnet, Instance, Reason_Code)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 46 | VALID-SERVICE-POLL-RES | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>=><br>MSAC2M_Reject.ind(C_Ref:=Stored_C_Ref, Reason_Code:=REJ_PS) | VALID-SERVICE-POLL-RES |
| 47 | VALID-SERVICE-POLL-RES | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE,<br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated, Subnet, Instance, Reason_Code) | VALID-SERVICE-POLL-RES |
| 48 | VALID-SERVICE-POLL_RES | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER | VALID-SERVICE-POLL-RES |
| 49 | VALID-SERVICE-POLL-RES | R-Timer expired<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_TO,<br>STORE_ABORT_PARAMETER<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated, Subnet, Instance, Reason_Code) | END-WAIT-CON |
| 50 | END-WAIT-CON | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/(L_status = DL && L_sdu <> Abort-REQ-PDU) \|\| (L_status = NR)<br>=><br>Stop S-Timer<br>Start S-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add,<br>L_sdu:=Abort-REQ-PDU with stored Parameter,<br>Serv_class:=Low) | END-POLL-RES |
| 51 | END-WAIT-CON | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DS/UE/RS/RDL/NA/RR/LR/RDH/DH<br>=><br>C_Ref:=Stored_C_Ref , Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=DLL, Reason_Code:=L_status<br>Stop S-Timer<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated, Subnet, Instance, Reason_Code) | CLOSED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 52 | END-WAIT-CON | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Abort-REQ-PDU<br>=><br>Stop S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=FALSE, Subnet, Instance, Reason_Code)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 53 | END-WAIT-CON | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE) | END-WAIT-CON |
| 54 | END-WAIT-CON | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE) | END-WAIT-CON |
| 55 | END-WAIT-CON | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE) | END-WAIT-CON |
| 56 | END-WAIT-CON | S-Timer expired<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code := ABT_OC) | FINAL-WAIT-CON |
| 57 | IDLE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = NR<br>=><br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu.len:=0, Serv_class:=Low) | IDLE-POLL-RES |
| 58 | IDLE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Idle-RES-PDU &&<br>Service_Buffer = empty && Go_Abort = FALSE<br>=><br>Stop R-Timer<br>Start S-Timer | OPEN |
| 59 | IDLE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Idle-RES-PDU &&<br>Service_Buffer <> empty && Go_Abort = FALSE<br>=><br>Service_Header:= Service_Buffer[1]<br>Service_Buffer:=empty<br>Stop R-Timer<br>Start R-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Service_Buffer, Serv_class:=Low) | VALID-SERVICE-POLL-RES |
| 60 | IDLE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Idle-RES-PDU && Go_Abort = TRUE<br>=><br>Stop R-Timer<br>Start S-Timer<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU with stored Parameter, Serv_class:=Low) | END-POLL-RES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 61 | IDLE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu = Abort-REQ-PDU<br>=><br>Stop R-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=FALSE, Subnet, Instance, Reason_Code)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 62 | IDLE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL && L_sdu <> Idle-RES-PDU && L_sdu <> Abort-REQ-PDU<br>=><br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_RE)<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU, Serv_class:=Low) | END-POLL-RES |
| 63 | IDLE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DS/UE/RS<br>=><br>Stop R-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=DLL, Reason_Code:=L_status)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 64 | IDLE-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP, Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = RDL/NA/RR/LR/DH/RDH<br>=><br>ignore L_sdu if present<br>Stop R-Timer<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=DLL, Reason_Code:=L_status)<br>DMPMM2_DATA_REPLY.req(SSAP:=Client_SAP, DSAP:=Stored_Server_SAP, Rem_Add:=Server_Add, L_sdu:=Abort-REQ-PDU, Serv_class:=Low) | END-POLL-RES |
| 65 | IDLE-POLL-RES | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>/Length <= (Stored_Max_Data_Length) &&<br>Length <= (Stored_Max_Local_Data_Length) &&<br>Service_Buffer = empty<br>=><br><LOAD_SERVICE_BUFFER> | IDLE-POLL-RES |
| 66 | IDLE-POLL-RES | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>/(Length > (Stored_Max_Data_Length) ||<br>Length > (Stored_Max_Local_Data_Length)) &&<br>Service_Buffer = empty<br>=><br>MSAC2M_Reject.ind(C_Ref:=Stored_C_Ref, Reason_Code:=REJ_LE) | IDLE-POLL-RES |
| 67 | IDLE-POLL-RES | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>//Service_Buffer <> empty<br>=><br>MSAC2M_Reject.ind(C_Ref:=Stored_C_Ref, Reason_Code:=REJ_PS) | IDLE-POLL-RES |
| 68 | IDLE-POLL-RES | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE,<br>Go_Abort := TRUE<br>STORE_ABORT_PARAMETER<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated, Subnet, Instance, Reason_Code) | IDLE-POLL-RES |
| 69 | IDLE-POLL_RES | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>Go_Abort:=TRUE<br>STORE_ABORT_PARAMETER | IDLE-POLL-RES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 70 | IDLE-POLL-RES | R-Timer expired<br>=><br>C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_TO,<br>STORE_ABORT_PARAMETER<br>Start S-Timer<br>MSAC2M_Abort.ind(C_Ref, Locally_Generated,<br>Subnet, Instance, Reason_Code) | END-WAIT-CON |
| 71 | END-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP,<br>Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = NR<br>=><br>Stop S-Timer<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 72 | END-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP,<br>Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DL<br>=><br>Stop S-Timer<br>ignore L_sdu<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 73 | END-POLL-RES | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP,<br>Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status = DS/UE/RS/RDL/NA/RR/LR/RDH/DH<br>=><br>ignore L_sdu if present<br>Stop S-Timer<br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=DLL, Reason_Code:=L_status)<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 74 | END-POLL-RES | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE) | END-POLL-RES |
| 75 | END-POLL-RES | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported,<br>Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_SE) | END-POLL-RES |
| 76 | END-POLL-RES | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>ignore | END-POLL-RES |
| 77 | END-POLL-RES | S-Timer expired<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code := ABT_OC) | FINAL-WAIT-CON |
| 78 | FINAL-WAIT-CON | DMPMM2_DATA_REPLY.cnf(SSAP=Client_SAP, DSAP=Stored_Server_SAP,<br>Rem_add=Server_Add, L_sdu, Serv_class=Low, L_Status)<br>/L_status <> IV/LS<br>=><br>ignore L_sdu if present<br>MSAC2M_Closed.ind(C_Ref:=Stored_C_Ref) | CLOSED |
| 79 | FINAL-WAIT-CON | MSAC2M_XXX.req(C_Ref=Stored_C_Ref, <REQPARAM>)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_OC) | FINAL-WAIT-CON |
| 80 | FINAL-WAIT-CON | MSAC2M_Initiate.req(C_Ref, Send_Timeout, Features_Supported,<br>Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE,<br>Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_OC) | FINAL-WAIT-CON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 81 | FINAL-WAIT-CON | MSAC2M_Abort.req(C_Ref=Stored_C_Ref, Subnet, Instance, Reason_Code)<br>=><br>MSAC2M_Abort.ind(C_Ref:=Stored_C_Ref, Locally_Generated:=TRUE, Subnet:=NO, Instance:=MSAC2M, Reason_Code:=ABT_OC) | FINAL-WAIT-CON |
| 82 | ANY-STATE | MSAC2M_Reset.req<br>=><br>Stop R-Timer<br>Stop S-Timer<br>MSAC2M_Reset.cnf | POWER-ON |

## 9.13 MMAC2

### 9.13.1 Primitive definitions

#### 9.13.1.1 Primitives exchanged between MMAC2 and FSPMM2

Table 99 shows the service primitives including their associated parameters issued by the FSPMM2 and received by the MMAC2.

**Table 99 – Primitives issued by FSPMM2 to MMAC2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit MM.req | FSPMM2 | Max Local Data Length | |
| Reset.req | FSPMM2 | (none) | |
| Abort.req | FSPMM2 | (none) | |
| Get Master Diag.req | FSPMM2 | Rem Add,<br>MDiag Identifier | |
| Start Seq.req | FSPMM2 | Rem Add,<br>Area Code,<br>Timeout | |
| Download.req | FSPMM2 | Rem Add,<br>Area Code,<br>Add Offset,<br>Data | |
| Upload.req | FSPMM2 | Rem Add,<br>Area Code,<br>Add Offset,<br>Data Len | |
| End Seq.req | FSPMM2 | Rem Add | |
| Act Param.req | FSPMM2 | Rem Add,<br>Area Code,<br>Activate | |
| Act Para Brct.req | FSPMM2 | Rem Add,<br>Area Code | |

Table 100 shows the service primitives including their associated parameters issued by the MMAC2 and received by the FSPMM2.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 100 – Primitives issued by MMAC2 to FSPMM2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit MM.cnf | MMAC2 | | |
| Reset.cnf | MMAC2 | (none) | |
| Get Master Diag.cnf(+) | MMAC2 | Rem Add,<br>Diagnosis Data | |
| Get Master Diag.cnf(-) | MMAC2 | Rem Add,<br>Status | |
| Start Seq.cnf(+) | MMAC2 | Rem Add,<br>Max Len Data Unit | |
| Start Seq.cnf(-) | MMAC2 | Rem Add,<br>Status | |
| Download.cnf(+) | MMAC2 | Rem Add | |
| Download.cnf(-) | MMAC2 | Rem Add,<br>Status | |
| Upload.cnf(+) | MMAC2 | Rem Add,<br>Data | |
| Upload.cnf(-) | MMAC2 | Rem Add,<br>Status | |
| End Seq.cnf(+) | MMAC2 | Rem Add | |
| End Seq.cnf(-) | MMAC2 | Rem Add,<br>Status | |
| Act Param.cnf(+) | MMAC2 | Rem Add | |
| Act Param.cnf(-) | MMAC2 | Rem Add,<br>Status | |
| Act Para Brct.cnf(+) | MMAC2 | Rem Add | |
| Act Para Brct.cnf(-) | MMAC2 | Rem Add,<br>Status | |
| Abort.ind | MMAC2 | Rem Add | |
| Fault.ind | MMAC2 | (none) | |

#### 9.13.1.2 Parameters of MMAC2 primitives

The parameters used with the primitives exchanged between MMAC2 and FSPMM2 are described in FAL Service Definition in IEC 61158-5-3. Additional parameters are described in Table 101.

**Table 101 – Parameters used with primitives exchanged with MMAC2**

| Parameter name | Description |
|---|---|
| Rem_Add | DL Address of the remote station |

### 9.13.2 State machine description

The state machine for Master-Master communication specifies the requester side. Typically, these are programming or diagnostic devices.

After reset, the MMAC2 changes to the “POWER-ON" state and waits for initialisation by the User.

The state machine resumes as long in the “WAIT-REQ" state as it receives a valid request from the User. After this request was sent, the state machine remains in the “WAIT-RES" state, until the service is terminated successfully or incorrectly.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

At any time, maximally one service request is in execution.

**Local Variables**

**T1**
(Unsigned16)

Time which can elapse at most between a request and the receipt of the associated response.

**Server_Add**
(Unsigned8)

Address of the DP-master (Class 1) just communicating with.

**Service-Header**
(Octet-String of length 4)

Intermediate memory for the first four octets of the last request.

**Srv_lst**
This list is used to hold all information which are necessary for the service SAP_ACTIVATE.

**Upload_Data_Len**
(Unsigned8)

This parameter specifies the length of the requested data.

### 9.13.3 MMAC2 state table

Table 102 contains the complete description of the MMAC2 state machine.

**Table 102 – MMAC2 state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | POWER-ON | MMAC2_Minit_MM.req<br>=>MMAC2_Minit_MM.cnf | WAIT-REQ |
| 2 | WAIT-REQ | MMAC2_Get_Master_Diag.req(Rem_Add,Identifier)<br>=>Service_Header[1]:=0x41<br>Service_Header[2]:=Identifier<br>Server_Add:=Rem_Add<br>Start T1<br>DMPMM2_DATA_REPLY.req(SSAP:=54, DSAP:=54, Rem_add:=Rem_Add,<br>L_sdu:=Service_Header, Serv_class:=Low) | WAIT-RES |
| 3 | WAIT-REQ | MMAC2_Start_Seq.req(Rem_Add,Area_Code,Timeout)<br>=>Service_Header[1]:=0x42<br>Service_Header[2]:=Area_Code<br>Service_Header[3-4]:=Timeout<br>Server_Add:=Rem_Add<br>Start T1<br>DMPMM2_DATA_REPLY.req(SSAP:=54, DSAP:=54, Rem_add:=Rem_Add,<br>L_sdu:=Service_Header, Serv_class:=Low) | WAIT-RES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 4 | WAIT-REQ | MMAC2_Download.req(Rem_Add,Area_Code,Add_Offset,Data)<br>=>Service_Header[1]:=0x42<br>Service_Header[2]:=Area_Code<br>Service_Header[3-4]:=Add_Offset<br>Server_Add:=Rem_Add<br>Start T1<br>DMPMM2_DATA_REPLY.req(SSAP:=54, DSAP:=54, Rem_add:=Rem_Add,<br>L_sdu:=(Service_Header,Data), Serv_class:=Low) | WAIT-RES |
| 5 | WAIT-REQ | MMAC2_Upload.req(Rem_Add,Area_Code,Add_Offset, Data_Len)<br>=>Service_Header[1]:=0x44<br>Service_Header[2]:=Area_Code<br>Service_Header[3-4]:=Add_Offset<br>Upload_Data_Len := Data_Len<br>Server_Add:=Rem_Add<br>Start T1<br>DMPMM2_DATA_REPLY.req(SSAP:=54, DSAP:=54, Rem_add:=Rem_Add,<br>L_sdu:=(Service_Header,Data_Len), Serv_class:=Low) | WAIT-RES |
| 6 | WAIT-REQ | MMAC2_End_Seq.req(Rem_Add)<br>=>Service_Header[1]:=0x45<br>Server_Add:=Rem_Add<br>Start T1<br>DMPMM2_DATA_REPLY.req(SSAP:=54, DSAP:=54, Rem_add:=Rem_Add,<br>L_sdu:=Service_Header[1], Serv_class:=Low) | WAIT-RES |
| 7 | WAIT-REQ | MMAC2_Act_Param.req(Rem_Add, Area_Code, Activate)<br>=>Service_Header[1]:=0x47<br>Service_Header[2]:=Area_Code<br>Service_Header[3]:=Activate<br>Server_Add:=Rem_Add<br>Start T1<br>DMPMM2_DATA_REPLY.req(SSAP:=54, DSAP:=54, Rem_add:=Rem_Add,<br>L_sdu:=Service_Header[1-3], Serv_class:=Low) | WAIT-RES |
| 8 | WAIT-REQ | MMAC2_Act_Para_Brct.req(Rem_Add, Area_Code)<br>=>DMPMM2_DATA.req(SSAP:=54, DSAP:=54, Rem_add:=Rem_Add,<br>L_sdu:=(0x46,Area_Code ), Serv_class:=Low) | WAIT-CON |
| 9 | WAIT-CON | MMAC2_XXX.req<br>=>MMAC2_Reject.ind(Rem_Add,Reason_Code:=REJ_CO) | WAIT-CON |
| 10 | WAIT-CON | DMPMM2_DATA.cnf(SSAP=54, DSAP=54, Rem_Add, Serv_Class, L_status)<br>/L_status=OK/DS<br>=>MMAC2_Act_Para_Brct.cnf(Rem_Add,Status:=L_status) | WAIT-REQ |
| 11 | WAIT-RES | MMAC2_XXX.req<br>=>MMAC2_Reject.ind(Rem_Add,Reason_Code:=REJ_PS) | WAIT-RES |
| 12 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class,<br>L_status)<br>/L_status=NR<br>=>DMPMM2_DATA_REPLY.req(SSAP:=54, DSAP:=54, Rem_add:=Server_Add,<br>L_sdu.len:=0, Serv_class:=Low) | WAIT-RES |
| 13 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class,<br>L_status)<br>/L_status=IV/LS<br>=>MMAC2_Fault.ind | POWER-ON |
| 14 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class,<br>L_status)<br>/L_status=DS/NA/UE/RR/RS && Service_Header[1]=0x41<br>=>Stop T1<br>MMAC2_Get_Master_Diag.cnf(Rem_Add, Status:=L_status,<br>Diagnostic_Data:=L_sdu) | WAIT-REQ |
| 15 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class,<br>L_status)<br>/L_status=DS/NA/UE/RR/RS && Service_Header[1]=0x42<br>=>Stop T1<br>MMAC2_Start_Seq.cnf(Rem_Add, Status:=L_status) | WAIT-REQ |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 16 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DS/NA/UE/RR/RS && Service_Header[1]=0x43<br>=>Stop T1<br>MMAC2_Download.cnf(Rem_Add, Status:=L_status) | WAIT-REQ |
| 17 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DS/NA/UE/RR/RS && Service_Header[1]=0x44<br>=>Stop T1<br>MMAC2_Upload.cnf(Rem_Add, Status:=L_status) | WAIT-REQ |
| 18 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DS/NA/UE/RR/RS && Service_Header[1]=0x45<br>=>Stop T1<br>MMAC2_End_Seq.cnf(Rem_Add, Status:=L_status) | WAIT-REQ |
| 19 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DS/NA/UE/RR/RS && Service_Header[1]=0x47<br>=>Stop T1<br>MMAC2_Act_Param.cnf(Rem_Add, Status:=L_status) | WAIT-REQ |
| 20 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=RDL/RDH && Service_Header[1]=0x41<br>=>Stop T1<br>MMAC2_Get_Master_Diag.cnf(Rem_Add, Status:=RR) | WAIT-REQ |
| 21 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=RDL/RDH && Service_Header[1]=0x42<br>=>Stop T1<br>MMAC2_Start_Seq.cnf(Rem_Add, Status:=RR) | WAIT-REQ |
| 22 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=RDL/RDH && Service_Header[1]=0x43<br>=>Stop T1<br>MMAC2_Download.cnf(Rem_Add, Status:=RR) | WAIT-REQ |
| 23 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=RDL/RDH && Service_Header[1]=0x44<br>=>Stop T1<br>MMAC2_Upload.cnf(Rem_Add, Status:=RR) | WAIT-REQ |
| 24 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=RDL/RDH && Service_Header[1]=0x45<br>=>Stop T1<br>MMAC2_End_Seq.cnf(Rem_Add, Status:=RR) | WAIT-REQ |
| 25 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=RDL/RDH && Service_Header[1]=0x47<br>=>Stop T1<br>MMAC2_Act_Param.cnf(Rem_Add, Status:=RR) | WAIT-REQ |
| 26 | WAIT-RES | T1 expired<br>/Service_Header[1]=0x41<br>=>MMAC2_Get_Master_Diag.cnf(Rem_Add, Status:=TO) | TIMO |
| 27 | WAIT-RES | T1 expired<br>/Service_Header[1]=0x42<br>=>MMAC2_Start_Seq.cnf(Rem_Add, Status:=TO) | TIMO |
| 28 | WAIT-RES | T1 expired<br>/Service_Header[1]=0x43<br>=>MMAC2_Download.cnf(Rem_Add, Status:=TO) | TIMO |
| 29 | WAIT-RES | T1 expired<br>/Service_Header[1]=0x44<br>=>MMAC2_Upload.cnf(Rem_Add, Status:=TO) | TIMO |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 30 | WAIT-RES | T1 expired<br>/Service_Header[1]=0x45<br>=>MMAC2_End_Seq.cnf(Rem_Add, Status:=TO) | TIMO |
| 31 | WAIT-RES | T1 expired<br>/Service_Header[1]=0x47<br>=>MMAC2_Act_Param.cnf(Rem_Add, Status:=TO) | TIMO |
| 32 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]=FE/NE/AD/EA/LE/RE/IP<br>&& Service_Header[1]=0x41<br>=>Stop T1<br>MMAC2_Get_Master_Diag.cnf(Rem_Add, Status:=L_sdu[1]) | WAIT-REQ |
| 33 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]=FE/NE/AD/EA/LE/RE/IP/NI/SE/SC<br>&& Service_Header[1]=0x42<br>=>Stop T1<br>MMAC2_Start_Seq.cnf(Rem_Add, Status:=L_sdu[1]) | WAIT-REQ |
| 34 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]=FE/NE/AD/EA/LE/RE/NC/SC/NI<br>&& Service_Header[1]=0x43<br>=>Stop T1<br>MMAC2_Download.cnf(Rem_Add, Status:=L_sdu[1]) | WAIT-REQ |
| 35 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]=FE/NE/AD/EA/LE/RE/NI/SC<br>&& Service_Header[1]=0x44<br>=>Stop T1<br>MMAC2_Upload.cnf(Rem_Add, Status:=L_sdu[1]) | WAIT-REQ |
| 36 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]=FE/NE/AD/EA/LE/RE/NI/SE<br>&& Service_Header[1]=0x45<br>=>Stop T1<br>MMAC2_End_Seq.cnf(Rem_Add, Status:=L_sdu[1]) | WAIT-REQ |
| 37 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]=FE/NE/AD/EA/LE/RE/IP/SC/NI/DI<br>&& Service_Header[1]=0x47<br>=>Stop T1<br>MMAC2_Act_Param.cnf(Rem_Add, Status:=L_sdu[1]) | WAIT-REQ |
| 38 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DH/LR \|\| L_status=DL<br>&& L_sdu[1]<>FE/NE/AD/EA/LE/RE/IP && Service_Header[1]=0x41<br>&& (L_sdu.len<3 \|\| L_sdu[1-2]<>Service_Header[1-2])<br>=>Stop T1<br>MMAC2_Get_Master_Diag.cnf(Rem_Add, Status:=RE) | WAIT-REQ |
| 39 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DH/LR \|\| L_status=DL<br>&& L_sdu[1]<>FE/NE/AD/EA/LE/RE/IP/NI/SE/SC<br>&& Service_Header[1]=0x42<br>&& (L_sdu.len<>5 \|\| L_sdu[1-4]<>Service_Header[1-4])<br>=>Stop T1<br>MMAC2_Start_Seq.cnf(Rem_Add, Status:=RE) | WAIT-REQ |
| 40 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DH/LR \|\| L_status=DL<br>&& L_sdu[1]<>FE/NE/AD/EA/LE/RE/NC/SC/NI<br>&& Service_Header[1]=0x43<br>&& (L_sdu[1-4]<>Service_Header[1-4] \|\| L_sdu.len<>4)<br>=>Stop T1<br>MMAC2_Download.cnf(Rem_Add, Status:=RE) | WAIT-REQ |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 41 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DH/LR \|\| L_status=DL<br>&& L_sdu[1]<>FE/NE/AD/EA/LE/RE/NI/SC<br>&& Service_Header[1]=0x44<br>&& (L_sdu.len<5 \|\| L_sdu[1-4]<>Service_Header[1-4]<br>\|\| L_sdu.len> Upload_Data_Len +4)<br>=>Stop T1<br>MMAC2_Upload.cnf(Rem_Add, Status:=RE) | WAIT-REQ |
| 42 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DH/LR \|\| L_status=DL<br>&& L_sdu[1]<>FE/NE/AD/EA/LE/RE/NI/SE<br>&& Service_Header[1]=0x45<br>&& (L_sdu[1]<>Service_Header[1] \|\| L_sdu.len<>1)<br>=>Stop T1<br>MMAC2_End_Seq.cnf(Rem_Add, Status:=RE) | WAIT-REQ |
| 43 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DH/LR \|\| L_status=DL<br>&& L_sdu[1]<>FE/NE/AD/EA/LE/RE/IP/SC/NI/DI<br>&& Service_Header[1]=0x47<br>&& (L_sdu[1-3]<>Service_Header[1-3] \|\| L_sdu.len<>3)<br>=>Stop T1<br>MMAC2_Act_Param.cnf(Rem_Add, Status:=RE) | WAIT-REQ |
| 44 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]<>FE/NE/AD/EA/LE/RE/IP<br>&& Service_Header[1]=0x41 && (L_sdu.len>=3<br>&& L_sdu[1-2]=Service_Header[1-2])<br>=>Stop T1<br>MMAC2_Get_Master_Diag.cnf(Rem_Add, Status:=OK,<br>Diagnostic_Data:=L_sdu[3-L_sdu.len]) | WAIT-REQ |
| 45 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]<>FE/NE/AD/EA/LE/RE/IP/NI/SE/SC<br>&& Service_Header[1]=0x42 && (L_sdu.len=5<br>&& L_sdu[1-4]=Service_Header[1-4])<br>=>Stop T1<br>MMAC2_Start_Seq.cnf(Rem_Add, Status :=OK,<br>Max_Length_Data_Unit:=L_sdu[5]) | WAIT-REQ |
| 46 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]<>FE/NE/AD/EA/LE/RE/NC/SC/NI<br>&& Service_Header[1]=0x43<br>&& (L_sdu[1-4]=Service_Header[1-4] && L_sdu.len=4)<br>=>Stop T1<br>MMAC2_Download.cnf(Rem_Add, Status:=OK) | WAIT-REQ |
| 47 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]<>FE/NE/AD/EA/LE/RE/NI/SC<br>&& Service_Header[1]=0x44 && (L_sdu.len>=5<br>&& L_sdu[1-4]=Service_Header[1-4]<br>&& L_sdu.len<= Upload_Data_Len +4)<br>=>Stop T1<br>MMAC2_Upload.cnf(Rem_Add, Status:=OK,Data:=L_sdu[5-L_sdu.len]) | WAIT-REQ |
| 48 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]<>FE/NE/AD/EA/LE/RE/NI/SE<br>&& Service_Header[1]=0x45<br>&& (L_sdu[1]=Service_Header[1] && L_sdu.len=1)<br>=>Stop T1<br>MMAC2_End_Seq.cnf(Rem_Add, Status:=OK) | WAIT-REQ |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 49 | WAIT-RES | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status)<br>/L_status=DL && L_sdu[1]<>FE/NE/AD/EA/LE/RE/IP/SC/NI/DI<br>&& Service_Header[1]=0x47<br>&& (L_sdu=Service_Header && L_sdu.len=3)<br>=>Stop T1<br>MMAC2_Act_Param.cnf(Rem_Add, Status:=OK) | WAIT-REQ |
| 50 | TIMO | DMPMM2_DATA_REPLY.cnf(SSAP=54, DSAP=54, Rem_Add, L_sdu, Serv_Class, L_status) | WAIT-REQ |
| 51 | TIMO | MMAC2_XXX.req<br>=>MMAC2_Reject.ind(Rem_Add,Reason_Code:=REJ_CO) | TIMO |
| 52 | any state | unexpected DL reaction<br>=>Stop T1<br>MMAC22_Fault.ind | POWER-ON |
| 53 | any state | MMAC2_Reset.req<br>=>Stop T1 | WAIT-REQ |

# 10 DLL mapping protocol machines (DMPMs)

## 10.1 DMPMS

### 10.1.1 Primitive definitions

#### 10.1.1.1 Primitives exchanged between DMPMS and FSPMS

Table 103 shows the service primitives including their associated parameters issued by the FSPMS and received by the DMPMS.

**Table 103 – Primitives issued by FSPMS to DMPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit DLL.req | FSPMS | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMS | (none) | |

Table 104 shows the service primitives including their associated parameters issued by the DMPMS and received by the FSPMS.

**Table 104 – Primitives issued by DMPMS to FSPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| SInit DL.cnf | DMPMS | Loc_Station_Address | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | DMPMS | (none) | |
| Fault.ind | DMPMS | (none) | |

#### 10.1.1.2 Primitives exchanged between DMPMS and MSCY1S

Table 105 shows the service primitives including their associated parameters issued by the MSCY1S and received by the DMPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 105 – Primitives issued by MSCY1S to DMPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Slave Init req | MSCY1S | (none) | Initialising of the resorces used by DP |
| Slave Deact.req | MSCY1S | (none) | Deactivation of LSAPs for MS0, MS1 |
| Enter.req | MSCY1S | (none) | Preparing the DLL resources for Data Exchange |
| Leave.req | MSCY1S | (none) | Release the DLL resources for Data Exchange |
| Set minTsdr.req | MSCY1S | MinTsdr | Change of the minTsdr Parameter in DLL |
| Slave Diag Upd.req | MSCY1S | Diag Data,<br>Reference | Setup of the DL update buffer for Diagnosis |
| Data Exchange Upd.req | MSCY1S | Diag Flag,<br>Inp Data | Transfer of the Inp Data buffer to the DL update buffer |
| RD Outp Upd.req | MSCY1S | Outp Data | Setup a copy of the Outp Data buffer |
| RD Inp Upd.req | MSCY1S | Inp Data | Setup a copy of the Inp Data buffer |
| Get Cfg Upd.req | MSCY1S | Cfg Data | Setting of configuration data |

Table 106 shows the service primitives including their associated parameters issued by the DMPMS and received by the MSCY1S.

**Table 106 – Primitives issued by DMPMS to MSCY1S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Slave Init.cnf | DMPMS | (none) | Success of Initialisation |
| Reset.cnf | DMPMS | (none) | Termination of Reset service |
| Slave Deact.cnf | DMPMS | (none) | Success of deactivation of LSAPs for MS0, MS1 |
| Set Slave Add.ind | DMPMS | New Slave Add,<br>Ident Number,<br>No Add Chg,<br>Rem Slave Data | Change of the DL address of a DP-slave. Setting of remanent Slave Data. |
| Slave Diag.ind | DMPMS | Req Add,<br>Reference | A DP-master fetches diagnosis information. |
| Set Prm.ind | DMPMS | Req Add,<br>Prm Data | Parameter data are delivered to the DP-slave. The parameterization is first done in the start-up phase and is also while the DP-slave is in the Data Exchang mode. |
| Chk Cfg.ind | DMPMS | Req Add,<br>Cfg Data | Received configuration data to the DP-slave for checking. They contain the format of input and output areas as well as the information about the data consistency. |
| Data Exchange.ind | DMPMS | Outp Data | Transmits output data to a DP-slave and requests input data. |
| Global Control.ind | DMPMS | Control Command,<br>Group Select | Control Command accepted only from DP master (Class1) to which this Slave is assigned.<br>Master reports his Operation Mode with this command. |

#### 10.1.1.3 Primitives exchanged between DMPMS and SSCY1S

Table 107 shows the service primitives including their associated parameters issued by the DMPMS and received by the SSCY1S.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 107 – Primitives issued by DMPMS to SSCY1S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DX_Entered.ind | DMPMS | Rem_add,<br>Status | entered the DATA-EXCH mode |
| DX_Broadcast.ind | DMPMS | Rem_add,<br>Data | Broadcast received |

#### 10.1.1.4 Primitives exchanged between DMPMS and MSAC1S, MSRM2S, MSAC2S

Table 108 shows the service primitives including their associated parameters issued by the MSAC1S, MSRM2S and MSAC2S and received by the DMPMS.

**Table 108 – Primitives issued by MSAC1S, MSRM2S, MSAC2S to DMPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| RSAP_ACTIVATE.req | MSAC1S,<br>MSRM2S,<br>MSAC2S | SSAP,<br>Access,<br>L_sdu_length_list,<br>Indication_Mode | Activates local SAP |
| SAP_DEACTIVATE.req | MSAC1S,<br>MSRM2S,<br>MSAC2S | SSAP | Deactivates local SAP |
| REPLY_UPDATE.req | MSAC1S,<br>MSRM2S,<br>MSAC2S | SSAP,<br>L_sdu,<br>Serv_class,<br>Transmit | Write Data in the Update Buffer for Transmission |

Table 109 shows the service primitives including their associated parameters issued by the DMPMS and received by the MSAC1S, MSRM2S and MSAC2S.

**Table 109 – Primitives issued by DMPMS to MSAC1S, MSRM2S, MSAC2S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| RSAP_ACTIVATE.cnf | DMPMS | SSAP,<br>M_Status | Activation finished |
| SAP_DEACTIVATE.cnf | DMPMS | SSAP,<br>M_Status | Deactivation finished |
| REPLY_UPDATE.cnf | DMPMS | SSAP,<br>Serv_class,<br>L_status | Update finished |
| DATA_REPLY.ind | DMPMS | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class,<br>Loc_Add,<br>Update_status | Data/Request received |

#### 10.1.1.5 Primitives exchanged between DMPMS and MSCS1S

Table 110 shows the service primitives including their associated parameters issued by DMPMS and received by the MSCS1S.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Table 110 – Primitives issued by DMPMS to MSCS1S**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| CS_CLOCK_VALUE.ind | DMPMS | Time_master_addr<br>CS_list,<br>CS_status,<br>Receive_delay_time | Indicates receiving of a clock message |

### 10.1.1.6 Primitives exchanged between DMPMS and DL

Table 111 shows the service primitives including their associated parameters issued by the DMPMS and received by the DL.

**Table 111 – Primitives issued by DMPMS to DL**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DL-REPLY-UPDATE.req | DMPMS | Service_class,<br>S_SAP_index,<br>DLSDU,<br>Transmit_strategy,<br>Reference | |
| DLM-RESET.req | DMPMS | (none) | |
| DLM-SET-VALUE.req | DMPMS | Variable_name(1 to n),<br>Desired_value (1 to n) | |
| DLM-DLSAP-ACTIVATE.req | DMPMS | S_SAP_index,<br>Access,<br>Service_list | |
| DLM-DLSAP-ACTIVATE-RESPONDER.req | DMPMS | S_SAP_index,<br>Access,<br>DLSDU_length_list,<br>Indication_mode,<br>Publisher_enabled | |
| DLM-DLSAP-ACTIVATE-SUBSCRIBER.req | DMPMS | S_SAP_index,<br>DLSDU_length_list | |
| DLM-DLSAP-DEACTIVATE.req | DMPMS | S_SAP_index | |

Table 112 shows the service primitives including their associated parameters issued by the DL and received by the DMPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 112 – Primitives issued by DL to DMPMS**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DL-DATA.ind | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_addr,<br>S_SAP_index,<br>DLSDU | Refer to DL Service Definition in IEC 61158-3-3 |
| DL-DATA-REPLY.ind | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_addr,<br>S_SAP_index,<br>DLSDU,<br>Update_Status<br>Reference | |
| DL-MCT-DATA-REPLY.ind | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_addr,<br>S_SAP_index,<br>DLSDU,<br>Update_Status<br>Reference | |
| DL-REPLY-UPDATE.cnf | DL | Service_class,<br>S_SAP_index,<br>DL_status | |
| DL-DXM-REPLY.ind | DL | D_addr,<br>D_SAP_index,<br>S_addr,<br>S_SAP_index,<br>DLSDU | |
| DL-CS-CLOCK-VALUE.ind | DL | D_addr,<br>D_SAP_index,<br>S_addr,<br>S_SAP_index,<br>DLSDU,<br>Receive_delay_time,<br>CS_Status | |
| DLM-RESET.cnf | DL | DLM_Status | |
| DLM-SET-VALUE.cnf | DL | Current_value (1 to n),<br>DLM_status | |
| DLM-DLSAP-ACTIVATE.cnf | DL | S_SAP_index,<br>DLM_status | |
| DLM-DLSAP-ACTIVATE-RESPONDER.cnf | DL | S_SAP_index,<br>DLM_status | |
| DLM-DLSAP-ACTIVATE-SUBSCRIBER.cnf | DL | S_SAP_index,<br>DLM_status | |
| DLM-DLSAP-DEACTIVATE.cnf | DL | S_SAP_index,<br>DLM_status | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010 Withdrawn

#### 10.1.1.7 Parameters of DMPMS primitives

The parameters used with the primitives exchanged between DMPMS and other state machines of the Slave AL are described in Table 113.

**Table 113 – Parameters used with primitives exchanged with DMPMS**

| Parameter name | Description |
|---|---|
| MasterAdd | This parameter conveys the DL address of the assigned DP-master. |
| Input Data Len | This parameter is used to setup the length of Input Data for this Slave. |
| Output Data Len | This parameter is used to setup the length of Output Data for this Slave. |
| Min Tsdr | This parameter conveys the value of the DLL variable minTsdr. |
| Diag Data | This parameter conveys the Diagnosis-RES-PDU |
| Diag Flag | This parameter indicates that a DP-slave has new Diag Data |
| Inp Data | This parameter conveys the DataExchange-RES-PDU |
| Outp Data | This parameter conveys the DataExchange-REQ-PDU |
| Cfg Data | This parameter conveys the Chk_Cfg-REQ-PDU |
| New Slave Add | This parameter conveys the new DL address for this DP-slave. |
| Ident Number | This parameter conveys the device type identifier for this DP-slave. |
| No Add Chg | This parameter indicates that there is no further address change allowed. |
| Rem Slave Data | This parameter conveys remanent parameter for this DP-slave. |
| Req Add | This parameter conveys the DL address of the initiator of this service. |
| Prm Data | This parameter conveys the Set_Prm-REQ-PDU. |
| Control Command | This parameter conveys the sync, freeze actions and the state of the DP-master. |
| Group Select | This parameter determines which groups shall be addressed by this command. |
| SSAP | Local Identifier for Service Access Point |
| Access | Defines the use of a SAP for one or several Remote Address (Stations) |
| L_sdu_length_list | Defines the maximum length of incoming or outgoing L_sdu |
| Indication_Mode | Defines the Mode for Indication of poll-sequences without user data |
| Serv_class | Indicates priority of the service |
| M_status | Status of the service execution |
| L_status | Status of the service execution |
| Transmit | Defines the operation mode of the update buffer (single = buffer is transmitted once) |
| Rem_Add | DL Address of the remote station |
| Loc_Add | DL Address of this station |
| Update_status | Indicates the transmission of an update buffer |
| Reference | Handle to identify diagnosis data |
| Data | This parameter conveys the data |
| Time_master_addr | DL address of the active Time Master |
| CS_list | List of Time Value at the transmission of this service (Clock_value_time_event), Time Value at the previous transmission (Clock_value_previous_TE) and status information related to the clock value (Clock_value_status) |
| CS_status | Indicates the success or failure of the Clock Synchronisation sequence |
| Receive_delay_time | Time expired between receipt of the Time_Event and the invocaton of that service primitive |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 10.1.2 State machine description

The DMPMS (DL Mapping Protocol Machine) makes the connection between the MS0-, MS1-, MS2-related Protocol Machines and Layer 2. All services are mapped by the DMPMS to the DL and its management services of Layer 2. The DMPMS delivers the necessary Layer 2 parameters of the function call (SSAP, DSAP, Serv_class, ...) for Layer 2, receives the confirmations and indications from Layer 2 and passes them to the MS0-, MS1-, MS2-related Protocol Machines.

**Local Variables**

**Loc_Station_Address**
(Unsigned8)

Local variable for the intermediate storage of the own station address which was delivered with the last Set_Value.

**Prev_Diag_Flag**
(Boolean)

Indicates the state of the Diag_Flag at the last Data_Exchange_Update.req By means of this flag it is detected whether a high prior update is present.

### 10.1.3 DMPMS state table

Table 114 contains the complete description of the DMPMS state machine.

**Table 114 – DMPMS state table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | PON | DMPMS_SInit_DLL.req(Bus Para)<br>=><br>Act_cnt := 0<br>DLM_SET_VALUE.req(<br>Variable_name1:= “TS “,<br>Variable_name2:= “Data_rate “,<br>Variable_name3:= “TSL “,<br>Variable_name4:= “min TSDR “,<br>Desired_Value1:= Bus_Para.TS<br>Desired_Value2:= Bus_Para.Data_rate,<br>Desired_Value3:= Bus_Para.TSL,<br>Desired_Value4:= 11) | DLM-SET-VALUE |
| 2 | DLM-SET-VALUE | DLM_SET_VALUE.cnf(DLM_status(1-5)<br>/DLM_status(1-4)=OK<br>=><br>no action<br>DMPMS_SInit_DLL.cnf | RUN |
| 3 | RUN | DMPMS_Enter.req<br>=><br>DMPMS_ActivateControl := DMPMS_Enter<br>Status:=TRUE<br>DMPMS_DX_Entered.ind (0..125, Status) | DLM-CHECK-RSAP-ACT |
| 4 | RUN | DMPMS_Slave_Init.req<br>/ActivateRSAPFromCRList(DMPMS_ActivateControl)=TRUE<br>=><br>DMPMS_ActivateControl := DMPMS_Init<br>DLSAP_ACTIVATE_RESPONDER.req( S_SAP_index:=CREP.DSAP,<br>Access:=CREP.Rem Add,<br>DLSDU_length_list:=BuildRSAPServiceListFromCR(),<br>Indication_mode:=CREP.Indication Mode, Publisher_enabled:=FALSE) | DLM-ACT-RSAP |
| 5 | DLM-CHECK-RSAP-ACT | /ActivateRSAPFromCRList(DMPMS_ActivateControl)=TRUE<br>=><br>DLSAP_ACTIVATE_RESPONDER.req( S_SAP_index:=CREP.DSAP,<br>Access:=CREP.Rem Add,<br>DLSDU_length_list:=BuildRSAPServiceListFromCR(),<br>Indication_mode:=CREP.Indication Mode, Publisher_enabled:=<br>CREP.Publisher_enabled ) | DLM-ACT-RSAP |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 6 | DLM-CHECK-RSAP-ACT | /ActivateRSAPFromCRList(DMPMS_ActivateControl)=FALSE && DMPMS_ActivateControl=DMPMS_Init<br>=><br>DMPMS_Slave_Init.cnf | RUN |
| 7 | DLM-CHECK-RSAP-ACT | /ActivateRSAPFromCRList(DMPMS_ActivateControl)=FALSE && DMPMS_ActivateControl=DMPMS_Enter<br>=><br>DLSAP_ACTIVATE.req(S_SAP_index:=58, Access:=CREP.Rem_Add, (Service_activate:=SDN, Role_in_service:=Responder, DLSDU_length_list:=(0,2,0,0))) | DLM-ACT-SAP |
| 8 | DLM-ACT-RSAP | DLSAP_ACTIVATE_RESPONDER.cnf(SSAP, DLM_status)<br>/DLM_status=OK | DLM-CHECK-RSAP-ACT |
| 9 | DLM-ACT-SAP | DLSAP_ACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=OK &&<br>EnableSubscriber() = FALSE<br>=><br>Prev_Diag_Flag := False | RUN |
| 10 | RUN | DMPMS_Leave.req<br>=><br>DMPMS_DeactControl:=DMPMS_Leave<br>Status:=FALSE<br>DMPMS_DX_Entered.ind (0..125, Status) | CHECK-SAP-DEACT |
| 11 | RUN | DMPMS_Slave_Deact.req<br>=><br>DMPMS_DeactControl:=DMPMS_Deact | CHECK-SAP-DEACT |
| 12 | CHECK-SAP-DEACT | /DeactivateSAPFromCRList(DMPMS_DeactControl)=TRUE<br>=><br>DLSAP_DEACTIVATE.req(S_SAP_index:=CREP.DSAP) | SAP-DEACT |
| 13 | CHECK-SAP-DEACT | /DeactivateSAPFromCRList(DMPMS_DeactControl)=FALSE<br>=><br>DMPMS_Slave_Deact.cnf | RUN |
| 14 | SAP-DEACT | DLSAP_DEACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=OK | CHECK-SAP-DEACT |
| 15 | RUN | DMPMS_Data_Exchange_Upd.req(Diag_Flag, Inp_Data)<br>/Diag_Flag = False && Prev_Diag_Flag = False && Inp_Data.len > 0<br>=><br>DL_REPLY_UPDATE.req(S:SAP_index:=NIL, DLSDU:=Inp_Data, Service_Class:=Low, Transmit_startegy:=Multiple, Reference:=Act_cnt) | RUN |
| 16 | RUN | DMPMS_Data_Exchange_Upd.req(Diag_Flag, Inp_Data)<br>/Diag_Flag = False && Prev_Diag_Flag = False && Inp_Data.len = 0<br>=><br>no action | RUN |
| 17 | RUN | DMPMS_Data_Exchange_Upd.req(Diag_Flag, Inp_Data)<br>/Diag_Flag = False && Prev_Diag_Flag = True && Inp_Data.len > 0<br>=><br>Prev_Diag_Flag := False<br>DL_REPLY_UPDATE.req(S_SAP_index:=NIL, DLSDU:=Inp_Data, Service_Class:=Low, Transmit_strategy:=Multiple, Reference:=Act_cnt)<br>DL_REPLY_UPDATE.req(S_SAP_index:=NIL, DLSDU:=NULL-PDU, Service_Class:=High, Transmit_strategy:=Single, Reference:=Act_cnt) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 18 | RUN | DMPMS_Data_Exchange_Upd.req(Diag_Flag, Inp_Data)<br>/Diag_Flag = False && Prev_Diag_Flag = True && Inp_Data.len = 0<br>=><br>Prev_Diag_Flag := False<br>DL_REPLY_UPDATE.req(S_SAP_index:=NIL, DLSDU:=NULL-PDU,<br>Service_Class:=High,<br>Transmit_strategy:=Single, Reference:=Act_cnt) | RUN |
| 19 | RUN | DMPMS_Data_Exchange_Upd.req(Diag_Flag, Inp_Data)<br>/Diag_Flag = True && Inp_Data.len > 0<br>=><br>Prev_Diag_Flag := True<br>DL_REPLY_UPDATE.req(S_SAP_index:=NIL, DLSDU:=Inp_Data,<br>Service_Class:=High,<br>Transmit_strategy:=Multiple, Reference:=Act_cnt) | RUN |
| 20 | RUN | DMPMS_Data_Exchange_Upd.req(Diag_Flag, Inp_Data)<br>/Diag_Flag = True && Inp_Data.len = 0<br>=><br>Prev_Diag_Flag := True<br>DL_REPLY_UPDATE.req(S_SAP_index:=NIL, DLSDU:=ZERO-PDU,<br>Service_Class:=High, Transmit_strategy:=Multiple, Reference:=Act_cnt) | RUN |
| 21 | RUN | DL_MCT_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Update_status, Reference)<br>/D_SAP_index=NIL<br>=><br>DMPMS_Data_Exchange.ind(Outp_Data:=DLSDU) | RUN |
| 22 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index=NIL<br>=><br>DMPMS_Data_Exchange.ind(Outp_Data:=DLSDU) | RUN |
| 23 | RUN | DMPMS_Slave_Diag_Upd.req(Diag_Data, Reference)<br>=><br>Act_cnt := Reference<br>DL_REPLY_UPDATE.req(S_SAP_index:=60, DLSDU:=Diag_Data,<br>Service_Class:=Low,<br>Transmit_strategy:=Multiple, Reference:=Act_cnt) | RUN |
| 24 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index=60<br>=><br>Act_cnt := Reference<br>DMPMS_Slave_Diag.ind(Req_Add:=S_addr, Reference := Act_cnt) | RUN |
| 25 | RUN | DMPMS_Get_Cfg_Upd.req(Cfg_Data)<br>=><br>DL_REPLY_UPDATE.req(S_SAP_index:=59, DLSDU:=Cfg_Data,<br>Service_class:=Low,<br>Transmit_strategy:=Multiple, Reference:=Act_cnt) | RUN |
| 26 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index=59<br>=><br>no action | RUN |
| 27 | RUN | DL_DATA.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU, Service_class)<br>/ D_SAP_index=58 && DLSDU = Global_Control-REQ-PDU<br>=><br>DMPMS_Global_Control.ind(<br>Control_Command:=Global_Control-REQ-PDU.Control_Command,<br>Group_Select:=Global_Control-REQ-PDU.Group_Select) | RUN |
| 28 | RUN | DL_DATA.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU, Service_class)<br>/ D_SAP_index=58 && DLSDU <> Global_Control-REQ-PDU<br>=><br>no action | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 29 | RUN | DMPMS_RD_Inp_Upd.req(Inp_Data)<br>/Inp_Data.len > 0<br>=><br>DL_REPLY_UPDATE.req(S_SAP_index:=56, DLSDU:=Inp_Data,<br>Service_Class:=Low,<br>Transmit_strategy:=Multiple, Reference:=Act_cnt) | RUN |
| 30 | RUN | DMPMS_RD_Inp_Upd.req(Inp_Data)<br>/Inp_Data.len = 0<br>=><br>no action | RUN |
| 31 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index=56<br>=><br>no action | RUN |
| 32 | RUN | DMPMS_RD_Outp_Upd.req(Outp_Data)<br>/Outp_Data.len > 0<br>=><br>DL_REPLY_UPDATE.req(S_SAP_index:=57, DLSDU:=Outp_Data,<br>Service_Class:=Low,<br>Transmit_strategy:=Multiple, Reference:=Act_cnt) | RUN |
| 33 | RUN | DMPMS_RD_Outp_Upd.req(Outp_Data)<br>/Outp_Data.len = 0<br>=><br>no action | RUN |
| 34 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index=57<br>=><br>no action | RUN |
| 35 | RUN | DMPMS_RSAP_ACTIVATE.req(SSAP, Access, L_sdu_length_list, Indication_Mode)<br>=><br>DLSAP_ACTIVATE_RESPONDER.req(S_SAP_index:=SSAP, Access:=Access,<br>DLSDU_length_list:=L_sdu_length_list, Indication_mode:=Indication_Mode,<br>Publisher_enabled:=FALSE) | DLM-PASS-ACT |
| 36 | DLM-PASS-ACT | DLSAP_ACTIVATE_RESPONDER.cnf(SSAP, DLM_status)<br>=><br>DMPMS_RSAP_ACTIVATE.cnf(SSAP:=S_SAP_index, M_Status:=DLM_status) | RUN |
| 37 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index<52<br>=><br>DMPMS_DATA_REPLY.ind(SSAP:=S_SAP_index, DSAP:=D_SAP_index,<br>Rem_Add:=D_addr, L_sdu:=DLSDU, Serv_class:=Service_class, Loc_Add:=S_addr,<br>Update_status:=Update_status) | RUN |
| 38 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index=55 && DLSDU=Set_Slave_Add-REQ-PDU<br>=><br>DMPMS_Set_Slave_Add.ind(<br>New_Slave_Add:=Set_Slave_Add-REQ-PDU.New_Slave_Add,<br>Ident_Number:=Set_Slave_Add-REQ-PDU.Ident_Number,<br>No_Add_Chg:=Set_Slave_Add-REQ-PDU.No_Add_Chg,<br>Rem_Slave_Data:=Set_Slave_Add-REQ-PDU.Rem_Slave_Data) | RUN |
| 39 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index=55 && DLSDU<>Set_Slave_Add-REQ-PDU<br>=><br>no action | RUN |
| 40 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU,<br>Service_class, Update_status, Reference)<br>/D_SAP_index=62<br>=><br>DMPMS_Chk_Cfg.ind(Req_Add:=S_addr, Cfg_Data:=DLSDU) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 41 | RUN | DMPMS_Set_minTsdr.req(minTsdr)<br>/MinTsdr = 0<br>=><br>no action | RUN |
| 42 | RUN | DMPMS_Set_minTsdr.req(minTsdr)<br>/minTsdr > 0<br>=><br>DLM_SET_VALUE.req(Variable_name1:=”minTSDR”, Desired_value1:=minTsdr) | SMINTSDR |
| 43 | SMINTSDR | DLM_SET_VALUE.cnf(DLM_status)<br>/DLM_Status=OK<br>=><br>no action | RUN |
| 44 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU, Service_class, Update_status, Reference)<br>/D_SAP_index=61<br>=><br>DMPMS_Set_Prm.ind(Req_Add:=S_add, Prm_Data:=DLSDU) | RUN |
| 45 | RUN | DMPMS_SAP_DEACTIVATE.req(SSAP)<br>=><br>DLSAP_DEACTIVATE.req(S_SAP_index:=SSAP) | DLM-PASS-DEACT |
| 46 | DLM-PASS-DEACT | DLSAP_DEACTIVATE.cnf( S_SAP_index, DLM_status)<br>=><br>DMPMS_SAP_DEACTIVATE.cnf(SSAP:=S_SAP_index, M_Status:=DLM_status) | RUN |
| 47 | RUN | DMPMS_REPLY_UPDATE.req(SSAP, L_sdu, Serv_class, Transmit)<br>/SSAP<52<br>=><br>DL_REPLY_UPDATE.req(S_SAP_index:=SSAP, DLSDU:=L_sdu, Service_class:=Serv_class, Transmit_strategy:=Transmit, Reference:=Act_cnt) | RUN |
| 48 | RUN | DL_REPLY_UPDATE.cnf(S_SAP_index, Service_class, DL_status)<br>/S_SAP_index<52<br>=><br>DMPMS_REPLY_UPDATE.cnf(SSAP:=S_SAP_index, Serv_class:=Service_class, L_status:=DL_status) | RUN |
| 49 | any state | DMPMS_Reset.req<br>=><br>DLM_RESET.req | WAIT-RESET |
| 50 | WAIT-RESET | DLM_RESET.cnf(DLM_status)<br>/DLM_status=OK<br>=><br>DMPMS_Reset.cnf | PON |
| 51 | RUN | DL_REPLY_UPDATE.cnf(S_SAP_index, Service_class, DL_status)<br>/(S_SAP_index=NIL \|\| S_SAP_index=62 \|\| S_SAP_index=61 \|\| S_SAP_index=60 \|\| S_SAP_index=59 \|\| S_SAP_index=57 \|\| S_SAP_index=56) && DL_status=OK<br>=><br>no action | RUN |
| 52 | any state | DL_CS_CLOCK_VALUE.ind (D_addr, D_SAP_index, S_addr, S_SAP_index, CS_list, CS_Status, Receive_delay_time)<br>=><br>CS_CLOCK_VALUE.ind (Time_master_addr := S_addr, CS_list, CS_Status, Receive Delay Time := Receive_delay_time) | SAME |
| 53 | any state | DLM_SET_VALUE.cnf(DLM_status)<br>/DLM_status=NO/IV<br>=><br>DMPMS_Fault.ind | PON |
| 54 | any state | DLSAP_ACTIVATE_RESPONDER.cnf(SSAP, DLM_status)<br>/DLM_status=NO/IV<br>=><br>DMPMS_Fault.ind | PON |
| 55 | any state | DLSAP_ACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=NO/IV<br>=><br>DMPMS_Fault.ind | PON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 56 | any state | DLSAP_DEACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=NO/IV<br>=><br>DMPMS_Fault.ind | PON |
| 57 | any state | DL_REPLY_UPDATE.cnf(S_SAP_index, Service_class, DL_status)<br>/DL_status=LS/LR/IV<br>=><br>DMPMS_Fault.ind | PON |
| 58 | any state | unexpected DL-primitive<br>/D_SAP_index>54<br>=><br>DMPMS_Fault.ind | PON |
| 59 | any state | unknown DL-primitive<br>/D_SAP_index>54<br>=><br>DMPMS_Fault.ind | PON |
| 60 | any state | unexpected DLM-primitive<br>=><br>DMPMS_Fault.ind | PON |
| 61 | any state | unknown DLM-primitive<br>=><br>DMPMS_Fault.ind | PON |
| 62 | RUN | DL_DXM_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU)<br>/IsFilterEntryExistent(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class)<br>=TRUE<br>=><br>Rem_Add:=Publisher Address<br>Data:=DLSDU<br>DMPMS_DX_Broadcast.ind (Rem_Add, Data) | RUN |
| 63 | RUN | DL_DXM_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU)<br>/IsFilterEntryExistent(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class)<br>=FALSE<br>=><br>no action | RUN |
| 64 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU, Service_class, Update_status, Reference)<br>/D_SAP_index=53<br>=><br>DMPMS_Set_Ext_Prm.ind(Req_Add:=S_add, Prm_Data:=DLSDU) | RUN |
| 65 | DLM-ACT-SAP | DLSAP_ACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=OK &&<br>EnableSubscriber() = TRUE<br>=><br>DLSAP_ACTIVATE_SUBSCRIBER.req(S_SAP_index:=NIL,<br>DLSDU_length_list:=(244,244)) | DLM-ACT-SUB |
| 66 | DLM-ACT-SAP | DLSAP_ACTIVATE_SUBSCRIBER.cnf( S_SAP_index, DLM_status)<br>/DLM_status=OK<br>=><br>Prev_Diag_Flag := False | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 10.1.4 Functions

Table 115 contains the functions used by the DMPMS, their arguments and their descriptions.

**Table 115 – Functions used by the DMPMS**

| Function name | Description |
|---|---|
| DeactivateSAPFromCRList (DMPMS_DeactControl) | This function checks the CRL DP-slave if there are CRL entries that belongs to MS0-AR and which SSAP has not been deactivated with the DLM-SAP-DEACTIVATE.req service primitive yet.<br>A) If DMPMS_DeactControl=DMPMS_Deact and if there exist one of the following CRL entries that has<br>not been deactivated yet:<br>Entries with DSAP=NIL or DSAP=58 or DSAP=57 or DSAP=56 or DSAP=62 or DSAP=61 or DSAP=60 or DSAP=59 or DSAP=55<br>then it returns TRUE and sets the local context according to the current CREP.<br>B) If DMPMS_DeactControl=DMPMS_Leave and if there exist one of the following CRL entries that has<br>not been deactivated yet:<br>Entries with DSAP=NIL or DSAP=58 or DSAP=57 or DSAP=56<br>then it returns TRUE and sets the local context according to one not activated CREP.<br>C) Otherwise it returns FALSE. |
| ActivateRSAPFromCRList (DMPMS_ActivateControl) | This function checks the CRL DP-slave if there are CRL entries that belongs to MS0-AR and which SSAP has not been activated with the DLM-RSAP-ACTIVATE.req service primitive yet.<br>A) If DMPMS_ActivateControl=DMPMS_Init and if there exist one of the following CRL entries that has<br>not been activated yet:<br>Entries with DSAP=62 or DSAP=61 or DSAP=60 or DSAP=59 or DSAP=55<br>then it returns TRUE and sets the local context according to the current CREP.<br>B) If DMPMS_ActivateControl=DMPMS_Enter and if there exist one of the following CRL entries that has<br>not been activated yet:<br>Entries with DSAP=NIL or DSAP=58 or DSAP=57 or DSAP=56<br>than it returns TRUE and sets the local context according to one not activated the current CREP.<br>C) Otherwise it returns FALSE. |
| BuildRSAPServiceListFrom CR() | This function builds the Service_list parameter for DL-RSAP-ACTIVATE.req primitive from the current CRL context.<br>It returns<br>DLSDU_length_list.Max_DLSDU_length_req_low = CREP.Max L_sdu Length Req Low<br>DLSDU_length_list.Max_DLSDU_length_req_high = CREP.Max L_sdu Length Req High<br>DLSDU_length_list.Max_DLSDU_length_ind/con_low =CREP.Max L_sdu Length Ind Low<br>DLSDU_length_list.Max_DLSDU_length_ind/con_high = CREP.Max L_sdu Length Ind High |
| IsFilterEntryExistent(S_SAP _index,<br>D_SAP_index,<br>S_addr,<br>DLSDU,<br>Service_class) | This function checks the CRL DP-slave if there are CRL entries that belongs to MS0-AR which contain DXB-Link entries .<br>A) If (S_SAP_Index && D_SAP_Index)=NIL and (Service_class=R-Brct) and if there exist one entry with<br>(DLSDU = Input Data Length Publisher && S_addr =Publisher Address):<br>than it returns TRUE and sets the local context according to the current CREP.<br>B) Otherwise it returns FALSE. |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Function name | Description |
|---|---|
| BuildServiceListFromCR() | This function builds the Service_list parameter for DLSAP-ACTIVATE.req primitive from the current CRL context.<br>A) If Enable Publisher = TRUE && EnableSubscriber() =FALSE it returns<br>Service_list.Service_activate[1] = SRD(R-Brct)<br>Service_list.Role_in_Service[1] = Initiator<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=0<br>B) If Enable Publisher = FALSE && EnableSubscriber() =TRUE it returns<br>Service_list.Service_activate[1] = SRD(R-Brct)<br>Service_list.Role_in_Service[1] = Responder<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=244<br>C)If Enable Publisher = TRUE && EnableSubscriber() =TRUE it returns<br>Service_list.Service_activate[1] = SRD(R-Brct)<br>Service_list.Role_in_Service[1] = Both<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=244 |
| EnableSubscriber() | This function checks the CRL DP-slave if there are CRL entries that belongs to MS0-AR which contain DXB-Link entries .<br>A) If there exist at least one entry:<br>than it returns TRUE and sets the local context according to the current CREP.<br>B) Otherwise it returns FALSE. |

## 10.2 DMPMM1

### 10.2.1 Primitive definitions

#### 10.2.1.1 Primitives exchanged between FSPMM1 and DMPMM1

Table 116 shows the service primitives including their associated parameters issued by the FSPMM1 and received by the DMPMM1.

**Table 116 – Primitives issued by FSPMM1 to DMPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit DLL.req | FSPMM1 | Bus Para | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.req | FSPMM1 | (none) | |
| Global Control.req | FSPMM1 | Control Command, Group Select | |
| Set Bus Par.req | FSPMM1 | Bus Para | |
| Delete SC.req | FSPMM1 | Address | |
| Read Value.req | FSPMM1 | Variable | |

Table 117 shows the service primitives including their associated parameters issued by the DMPMM1 and received by the FSPMM1.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 117 – Primitives issued by DMPMM1 to FSPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit DLL.cnf | DMPMM1 | (none) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | DMPMM1 | (none) | |
| Global Control.cnf(+) | DMPMM1 | (none) | |
| Global Control.cnf(-) | DMPMM1 | Status | |
| Set Bus Par.cnf | DMPMM1 | Status | |
| Delete SC.cnf | DMPMM1 | Status | |
| Read Value.cnf | DMPMM1 | Value, Status | |
| Fault.ind | DMPMM1 | (none) | |
| Event.ind | DMPMM1 | Event,<br>Add Info | |

#### 10.2.1.2 Primitives exchanged between MSCY1M and DMPMM1

Table 118 shows the service primitives including their associated parameters issued by the MSCY1M and received by the DMPMM1.

**Table 118 – Primitives issued by MSCY1M to DMPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Slave Diag.req | MSCY1M | Rem Add | Fetch Diagnosis of a DP-slave |
| Set Prm.req | MSCY1M | Rem Add,<br>Prm Data | Parameter data are delivered to the DP-slave. The parameterization is first done in the start-up phase and also while the DP-slave is in the Data Exchang mode. |
| Set ExtPrm.req | MSCY1M | Rem Add,<br>Prm Data | Parameter data are delivered to the DP-slave. The parameterization is first done in the start-up phase and also while the DP-slave is in the Data Exchang mode. |
| Chk Cfg.req | MSCY1M | Rem Add,<br>Cfg Data | Send configuration data to the DP-slave for checking. They contain the format of input and output areas as well as the information about the data consistency. |
| Data Exchange.req | MSCY1M | Rem Add,<br>Outp Data | Transmits output data to a DP-slave and requests input data. |

Table 119 shows the service primitives including their associated parameters issued by the DMPMM1 and received by the MSCY1M.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 119 – Primitives issued by DMPMM1 to MSCY1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| Slave Diag.cnf(+) | DMPMS | Rem Add,<br>Diag Data | Get Diagnosis from a DP-slave. |
| Slave Diag.cnf(-) | DMPMS | Rem Add,<br>Status | This primitive is used to convey a Slave Diag negative confirmation to DMPMM.. |
| Set Prm.cnf(+) | DMPMS | Rem Add | Parametrisation successfully transmitted, the parameters are not checked. |
| Set Prm.cnf(-) | DMPMS | Rem Add,<br>Status | This primitive is used to convey a Set Prm (Parameter Setting) negative confirmation to DMPMM. |
| Set ExtPrm.cnf(+) | DMPMS | Rem Add | Parametrisation successfully transmitted, the parameters are not checked. |
| Set ExtPrm.cnf(-) | DMPMS | Rem Add,<br>Status | This primitive is used to convey a Set ExtPrm (Parameter Setting) negative confirmation to DMPMM. |
| Chk Cfg.cnf(+) | DMPMS | Rem Add | Configuration data successfully transmitted, the parameters are not checked. |
| Chk Cfg.cnf(-) | DMPMS | Rem Add,<br>Status | This primitive is used to convey a Chk Cfg (configuration Check) negative confirmation to DMPMM. |
| Data Exchange.cnf(+) | DMPMS | Rem Add,<br>Diag Flag,<br>Inp Data | Get Input Data and Diag lag from a DP-slave |
| Data Exchange.cnf(-) | DMPMS | Rem Add,<br>Status | This primitive is used to convey a Data Exchange negative confirmation to DMPMS. |

### 10.2.1.3 Primitives exchanged between DMPMM1and MSAL1M, MSAC1M

Table 120 shows the service primitives including their associated parameters issued by MSAL1M and MSAC1M and received by the DMPMM1.

**Table 120 – Primitives issued by MSAL1M, MSAC1M to DMPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DATA_REPLY.req | MSAL1M,<br>MSAC1M | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class | Send Service |

Table 121 shows the service primitives including their associated parameters issued by the DMPMM1 and received by the MSAL1M and MSAC1M.

**Table 121 – Primitives issued by DMPMM1 to MSAL1M, MSAC1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DATA_REPLY.cnf | DMPMM1 | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class,<br>L_status | Send Service executed |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

#### 10.2.1.4 Primitives exchanged between DMPMM1 and MMAC1

Table 122 shows the service primitives including their associated parameters issued by MMAC1 and received by the DMPMM1.

**Table 122 – Primitives issued by MMAC1 to DMPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| RSAP_ACTIVATE.req | MMAC1 | SSAP,<br>Access,<br>L_sdu_length_list,<br>Indication_Mode | Activates local SAP |
| REPLY_UPDATE.req | MMAC1 | SSAP,<br>L_sdu,<br>Serv_class,<br>Transmit | Write Data in the Update Buffer for Transmission |

Table 123 shows the service primitives including their associated parameters issued by the DMPMM1 and received by the MMAC1.

**Table 123 – Primitives issued by DMPMM1 to MMAC1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| RSAP_ACTIVATE.cnf | DMPMM1 | SSAP,<br>M_Status | Activation finished |
| REPLY_UPDATE.cnf | DMPMM1 | SSAP,<br>Serv_class,<br>L_status | Update finished |
| DATA.ind | DMPMM1 | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class, | Data/Request received |
| DATA_REPLY.ind | DMPMM1 | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class,<br>Update_status | Data/Request received |

#### 10.2.1.5 Primitives exchanged between DMPMM1 and MSCS1M

Table 124 shows the service primitives including their associated parameters issued by MSCS1M and received by the DMPMM1.

**Table 124 – Primitives issued by MSCS1M to DMPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| CS_TIME_EVENT.req | FSPMM1 | (none) | Refer to DL Service Definition in IEC 61158-3-3 |
| CS_CLOCK_VALUE.req | FSPMM1 | CS_list | |

Table 125 shows the service primitives including their associated parameters issued by the DMPMM1 and received by the MSCS1M.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 125 – Primitives issued by DMPMM1 to MSCS1M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| CS_CLOCK_VALUE.ind | DMPMM1 | Time_master_addr<br>CS_list<br>Receive Delay Time<br>CS_status | Refer to DL Service Definition in IEC 61158-3-3 |
| CS_TIME_EVENT.cnf | DMPMM1 | Send_delay_time,<br>Status | |
| CS_CLOCK_VALUE.cnf | DMPMM1 | Status | |

### 10.2.1.6 Primitives exchanged between DMPMM1 and DL

Table 126 shows the service primitives including their associated parameters issued by the DMPMM1 and received by the DL.

**Table 126 – Primitives issued by DMPMM1 to DL**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DL-DATA.req | DMPMM1 | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DLSDU | Refer to DL Service Definition in IEC 61158-3-3 |
| DL-DATA-REPLY.req | DMPMM1 | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DLSDU | |
| DL-REPLY-UPDATE.req | DMPMM1 | Service_class,<br>S_SAP_index,<br>DLSDU,<br>Transmit_strategy | |
| DLM-RESET.req | DMPMM1 | (none) | |
| DLM-SET-VALUE.req | DMPMM1 | Variable_name(1 to n),<br>Desired_value (1 to n) | |
| DLM-GET-VALUE.req | DMPMM1 | Variable_name(1 to n) | |
| DLSAP-ACTIVATE.req | DMPMM1 | S_SAP_index,<br>Access,<br>Service_list | |
| DLSAP-ACTIVATE-RESPONDER.req | DMPMM1 | S_SAP_index,<br>Access,<br>DLSDU_length_list,<br>Indication_mode,<br>Publisher_enabled | |
| DLSAP-DEACTIVATE.req | DMPMM1 | S_SAP_index | |
| DL-CS_TIME_EVENT.req | DMPMM1 | D_addr,<br>D_SAP_index,<br>S_SAP_index | |
| DL-CS_CLOCK_VALUE.req | DMPMM1 | D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DLSDU | |

Table 127 shows the service primitives including their associated parameters issued by the DL and received by the DMPMM1.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 127 – Primitives issued by DL to DMPMM1**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DL-DATA.cnf | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DL_status | Refer to DL Service Definition in IEC 61158-3-3 |
| DL-DATA-REPLY.cnf | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DLSDU,<br>DL_status | |
| DL-DATA.ind | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_addr,<br>S_SAP_index,<br>DLSDU | |
| DL-DATA-REPLY.ind | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_addr,<br>S_SAP_index,<br>DLSDU,<br>Update_Status | |
| DL-REPLY-UPDATE.cnf | DL | Service_class,<br>S_SAP_index,<br>DL_status | |
| DLM-RESET.cnf | DL | DLM_Status | |
| DLM-SET-VALUE.cnf | DL | Current_value (1 to n),<br>DLM_status | |
| DLM-GET-VALUE.cnf | DL | Current_value (1 to n),<br>DLM_status (1 to n) | |
| DLM-EVENT.ind | DL | Event/Fault | |
| DLSAP-ACTIVATE.cnf | DL | S_SAP_index,<br>DLM_status | |
| DLSAP-ACTIVATE-RESPONDER.cnf | DL | S_SAP_index,<br>DLM_status | |
| DLSAP-DEACTIVATE.cnf | DL | S_SAP_index,<br>DLM_status | |
| DL-CS_CLOCK_VALUE.ind | DL | D_addr,<br>D_SAP_index,<br>S_addr,<br>S_SAP_index,<br>DLSDU,<br>Receive_delay_time,<br>CS_Status | |
| DL-CS_TIME_EVENT.cnf | DL | Send_delay_time,<br>Status | |
| DL-CS_CLOCK_VALUE.cnf | DL | Status | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 10.2.1.7 Parameters of DMPMM1 primitives

The parameters used with the primitives exchanged between DMPMM1 and the othe state machines of the Master (Class 1) AL are described in Table 128.

**Table 128 – Parameters used with primitives exchanged with DMPMM1**

| Parameter name | Description |
|---|---|
| Input Data Len | This parameter is used to setup the length of Input Data for this Slave. |
| Output Data Len | This parameter is used to setup the length of Output Data for this Slave. |
| Min Tsdr | This parameter conveys the value of the DLL variable minTsdr. |
| Rem Add | This parameter conveys the DL address of the receiver of this service. |
| Diag Data | This parameter conveys the Diagnosis-RES-PDU |
| Diag Flag | This parameter indicates that a DP-slave has new Diag Data |
| Prm Data | This parameter conveys the Set_Prm-REQ-PDU. |
| Cfg Data | This parameter conveys the Chk_Cfg-REQ-PDU |
| Inp Data | This parameter conveys the DataExchange-RES-PDU |
| Outp Data | This parameter conveys the DataExchange-REQ-PDU |
| Status | This parameter contains the cause of a negative confirmation- |
| SSAP | Local Identifier for Service Access Point |
| DSAP | Identifier for the remote Service Access Point |
| Access | Defines the use of a SAP for one or several Remote Address (Stations) |
| L_sdu_length_list | Defines the maximum length of incoming or outgoing L_sdu |
| Indication_Mode | Defines the Mode for Indication of poll-sequences without user data |
| Serv_class | Indicates priority of the service |
| M_status | Status ofthe service execution |
| L_status | Status ofthe service execution |
| L_sdu | Defines the DP PDU to be transmitted |
| Transmit | Defines the operation mode of the update buffer (single = buffer is transmitted once) |
| Rem_Add | DL Address of the remote station |
| Update_status | Indicates the transmission of an update buffer |
| Time_master_addr | DL address of the active Time Master |
| CS_list | List of Time Value at the transmission of this service (Clock_value_time_event), Time Value at the previous transmission (Clock_value_previous_TE) and status information related to the clock value (Clock_value_status) |
| CS_status | Indicates the success or failure of the Clock Synchronisation sequence |
| Receive_delay_time | Time expired between receipt of the Time_Event and the invocaton of that service primitive |

Table 129 shows the possible values of the parameter Status and the corresponding source entities.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 129 – Possible values of status**

| Value | Meaning | Source |
|---|---|---|
| DS | Local-DLL/PHY Entity is not in logical token-ring or disconnected from line | Local DLL |
| NA | Negative ack, no reaction from remote station | Local DLL |
| NR | No Response data | Remote DLL |
| OK | Acknowledgement positive | Local DLL<br>Remote DLL |
| RE | Format-Error in a Response-DLPDU | Local DMPM |
| RR | Resources of the remote-DLL Entity not sufficient or not available | Remote DLL |
| RS | Service or remote-address at remote-LSAP<br>or remote-LSAP not activated;<br>remote-station is no DP-Station<br>remote-station is not yet ready for these functions<br>remote-station is associated with an other Requester<br>optional service not available | Remote DLL |
| UE | Remote-MMAC/DLL interface error | Remote DLL |

### 10.2.2 State machine description

The DMPMM1 makes the connection between the MS0,MS1,MM1,MM2 specific parts and Layer 2. Services are mapped by the DMPMM1 to the DL and management of Layer 2. The DMPMM1 delivers the necessary Layer 2 parameters of the function call (SSAP, DSAP, Serv_class, ...) for Layer 2, receives the confirmations and indications from Layer 2 and passes them to the MS0, MS1, MM1/2-Handler.

**Local Variables**

**Act_Max_Diag_Len**
(Unsigned8)

Contains the maximal length of the diagnostic information which can be stored locally.

Range: 6 .. 244

**Loc_Station_Address**
(Unsigned8)

Local variable for the intermediate storage of the own station address which has been transferred with the last Set_Bus_Par.

Range: 0 .. 125

**Srv_lst**
This list is used to store all information which are necessary for the service SAP_ACTIVATE.

**Act_msac1m_Max_L_sdu_len**
(Unsigned8)

This variable is used to store the actual value of the parameter Max_L_sdu_len locally.

Range: 0 .. 240

### 10.2.3 DMPMM1 state table

Table 130 contains the complete description of the DMPMM1 state machine.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 130 – DMPMM1 state table**

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 1 | PON | DMPMM1_Minit_DLL.req(Bus_Para)<br>=><br>if (Isochronous Mode = Not Synchronized)<br>IsoM On:=FALSE<br>DLSDU:= NIL-SDU<br>else<br>IsoM On:=TRUE<br>DL IsoM On:=FALSE<br>DLSDU:=Global_Control-REQ-PDU(Control_Command:= 2, Group_Select:=Group_8)<br>endif<br><br>DLM_SET_VALUE.req(MSAP_0 ,<br>Variable_name1:= "TS ",<br>Variable_name2:= "Baud_rate ",<br>Variable_name3:= "TSL ",<br>Variable_name4:= "min TSDR ",<br>Variable_name5:= "max TSDR ",<br>Variable_name6:= "TQUI ",<br>Variable_name7:= "TSET ",<br>Variable_name8:= "TTR ",<br>Variable_name9:= "G ",<br>Variable_name10:= "HSA ",<br>Variable_name11:= "max_retry_limit ",<br>Variable_name12:= "in_ring_desired",<br>Variable_name13:= "SYNCHT",<br>Variable_name14:="maxTsh",<br>Variable_name15:="Tct",<br>Desired_Value1:= Bus_Para.TS,<br>Desired_Value2:= Bus_Para.Baud_rate,<br>Desired_Value3:= Bus_Para.TSL,<br>Desired_Value4:= Bus_Para.min TSDR,<br>Desired_Value5:= Bus_Para.max TSDR,<br>Desired_Value6:= Bus_Para.TQUI,<br>Desired_Value7:= Bus_Para.TSET,<br>Desired_Value8:= Bus_Para.TTR,<br>Desired_Value9:= Bus_Para.G,<br>Desired_Value10:= Bus_Para.HSA,<br>Desired_Value11:= Bus_Para.max_retry_limit,<br>Desired_Value12:= TRUE,<br>Desired_Value13:= DLSDU,<br>Desired_Value14:= maxTsh,<br>Desired_Value15:= Tct) | SET-VALUE |
| 2 | PON | DLM_RESET.cnf( DLM_status)<br>/DLM_status=OK<br>=><br>DMPMM1_Reset.cnf | PON |
| 3 | SET-VALUE | DLM_SET_VALUE.cnf( DLM_status(1-15))<br>/DLM_status(1-15)=OK<br>=><br>DLSAP_ACTIVATE.req( S_SAP_index:=SSAP, Access:=All,<br>Service_list:=BuildServiceListFromCR()) | SAP-ACT |
| 4 | SAP-ACT | DLSAP_ACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=OK && ActivateSAPFromCRList()=TRUE<br>=><br>DLSAP_ACTIVATE.req( S_SAP_index:=SSAP, Access:=All,<br>Service_list:=BuildServiceListFromCR()) | SAP-ACT |
| 5 | SAP-ACT | DLSAP_ACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=OK && ActivateSAPFromCRList()=FALSE<br>=><br>DLSAP_ACTIVATE_RESPONDER.req(S_SAP_index:=54, Access:=All,<br>DLSDU_length_list:=BuildLengthListFromCR()) | RSAP-ACT |
| 6 | RSAP-ACT | DLSAP_ACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=OK<br>=><br>DMPMM2_Minit_DLL.cnf | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 7 | RUN | DMPMM1_Set_Bus_Par.req(Bus_Para)<br>=><br>if (Isochron Mode = Not Synchronized)<br>IsoM On:=FALSE<br>DLSDU:= NIL-SDU<br>else<br>IsoM On:=TRUE<br>DL IsoM On:=FALSE<br>DLSDU:=Global_Control-REQ-PDU(Control_Command:= 2, Group_Select:=Group_8)<br>endif<br><br>DLM_SET_VALUE.req(MSAP_0 ,<br>Variable_name1:= "TS ",<br>Variable_name2:= "Baud_rate ",<br>Variable_name3:= "TSL ",<br>Variable_name4:= "min TSDR ",<br>Variable_name5:= "max TSDR ",<br>Variable_name6:= "TQUI ",<br>Variable_name7:= "TSET ",<br>Variable_name8:= "TTR ",<br>Variable_name9:= "G ",<br>Variable_name10:= "HSA ",<br>Variable_name11:= "max_retry_limit ",<br>Variable_name12:= "in_ring_desired",<br>Variable_name13:= "SYNCHT",<br>Variable_name14:="maxTsh",<br>Variable_name15:="Tct",<br>Desired_Value1:= Bus_Para.TS,<br>Desired_Value2:= Bus_Para.Baud_rate,<br>Desired_Value3:= Bus_Para.TSL,<br>Desired_Value4:= Bus_Para.min TSDR,<br>Desired_Value5:= Bus_Para.max TSDR,<br>Desired_Value6:= Bus_Para.TQUI,<br>Desired_Value7:= Bus_Para.TSET,<br>Desired_Value8:= Bus_Para.TTR,<br>Desired_Value9:= Bus_Para.G,<br>Desired_Value10:= Bus_Para.HSA,<br>Desired_Value11:= Bus_Para.max_retry_limit,<br>Desired_Value12:= TRUE,<br>Desired_Value13:= DLSDU,<br>Desired_Value14:= maxTsh,<br>Desired_Value15:= Tct) | SV1 |
| 8 | SV1 | DLM_SET_VALUE.cnf( DLM_status(1-15))<br>/DLM_status(1-15)=OK<br>=><br>DMPMM1_Set_Bus_Par.cnf(+) | RUN |
| 9 | RUN | DMPMM1_Delete_SC.req(Address)<br>=><br>DLM_SET_VALUE.req( Variable_name1:="DLPDU_Sent_Count",<br>Index1:=Address,<br>Variable_name2:="Error_Count",<br>Index2:=Address,<br>Desired_Value1:=0,<br>Desired_Value2:=0) | SV2 |
| 10 | SV2 | DLM_SET_VALUE.cnf( DLM_status(1-2))<br>/DLM_status(1-2)=OK<br>=><br>DMPMM1_Delete_SC.cnf(+) | RUN |
| 11 | RUN | DMPMM1_Read_Value.req( Variable)<br>=><br>DLM_GET_VALUE.req(Variable_name1:=Variable) | RUN |
| 12 | RUN | DLM_GET_VALUE.cnf( Current_value(1), DLM_status(1))<br>/DLM_status(1)=OK/NO<br>=><br>DMPMM1_Read_Value.cnf(Status:=DLM_status(1),Value:=Current_value(1)) | RUN |
| 13 | RUN | DLM_EVENT.ind( Event/Fault, Tsh)/Event/Fault<>Synch_Delay<br>&&Event/Fault<>Synch=>DMPMM1_Event.ind(Event:=Event/Fault) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 14 | RUN | DLM_EVENT.ind( Event/Fault, Tsh)<br>/Event/Fault=Synch_Delay<br>=><br>DMPMM1_SYNCH_Delayed.ind(TSH:=Tsh) | RUN |
| 15 | RUN | DLM_EVENT.ind( Event/Fault, Tsh)<br>/Event/Fault=Synch<br>=><br>DMPMM1_SYNCH.ind | RUN |
| 16 | RUN | DMPMM1_RSAP_ACTIVATE.req(SSAP, Access, L_sdu_length_list, Indication_Mode)<br>=><br>DLSAP_ACTIVATE_RESPONDER.req(S_SAP_index:=SSAP, Access:=Access, DLSDU_length_list:= L_sdu_length_list, Indication_mode:= Indication_Mode) | RUN |
| 17 | RUN | DLSAP_ACTIVATE_RESPONDER.cnf(SSAP, DLM_status)<br>/SSAP = 54 && DLM_status=OK<br>=><br>DMPMM1_RSAP_ACTIVATE.cnf(SSAP:=SSAP,DLM_status) | RUN |
| 18 | RUN | DMPMM1_REPLY_UPDATE.req(SSAP, L_sdu, Serv_class, Transmit)<br>=><br>DL_REPLY_UPDATE.req(S_SAP_index:=SSAP, DLSDU:=L_sdu, Service_class:=Serv_class, Transmit_strategy:=Transmit) | RUN |
| 19 | RUN | DL_REPLY_UPDATE.cnf(S_SAP_index, Service_class, DL_status)<br>=><br>DMPMM1_REPLY_UPDATE.cnf (SSAP:=S_SAP_index, Serv_class:=Service_class, L_status:=DL_status ) | RUN |
| 20 | RUN | DMPMM1_DATA_REPLY.req (SSAP, DSAP, Rem_Add, L_sdu, Serv_class )<br>=><br>DL_DATA_REPLY.req (S_SAP_index:=SSAP, D_SAP_index:=DSAP, S_addr:=Rem_Add, DLSDU:=L_sdu, Service_class:=Serv_class ) | RUN |
| 21 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index<>62 && S_SAP_index<>NIL && DL_status<>LS/IV<br>=><br>DMPMM1_DATA_REPLY.cnf (SSAP:=S_SAP_index, DSAP:=D_SAP_index, Rem_Add:=S_addr, L_sdu:=DLSDU, Serv_class:=Service_class, L_status:=DL_status ) | RUN |
| 22 | RUN | DL_DATA.ind(S_SAP_index, D_SAP_index, D_addr, S_Addr, DLSDU, Service_class)<br>=><br>DMPMM1_DATA.ind (SSAP:=S_SAP_index, DSAP:=D_SAP_index, Rem_Add:=S_addr, L_sdu:=DLSDU, Serv_class:=Service_class) | RUN |
| 23 | RUN | DL_DATA_REPLY.ind(S_SAP_index, D_SAP_index, S_addr, D_addr, DLSDU, Service_class, Update_status)<br>=><br>DMPMM1_DATA_REPLY.ind(SSAP:=S_SAP_index, DSAP:=D_SAP_index, Rem_Add)=S_addr, L_sdu:=DLSDU, Serv_class:=Service_class, Update_status:=Update_status) | RUN |
| 24 | RUN | DMPMM1_Data_Exchange.req(Rem_Add, Outp_Data)<br>/GetCRAttribute(PublisherFlag) = TRUE<br>=><br>DL_MCT_DATA_REPLY.req(S_SAP_index:=NIL, D_SAP_index:=NIL, D_addr:=Rem_Add, DLSDU:=Outp_Data, Service_class:=High) | RUN |
| 25 | RUN | DMPMM1_Data_Exchange.req(Rem_Add, Outp_Data)<br>/GetCRAttribute(PublisherFlag)=FALSE<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=NIL, D_SAP_index:=NIL, D_addr:=Rem_Add,DLSDU:=Outp_Data, Service_class:=High) | RUN |
| 26 | RUN | DL_MCT_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=NR<br>=><br>DMPMM1_Data_Exchange.cnf(+)(Rem_Add:=S_addr, Diag_Flag:=False, Inp_Data.len:=0) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 27 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=NR<br>=><br>DMPMM1_Data_Exchange.cnf(+)(Rem_Add:=S_addr, Diag_Flag:=False, Inp_Data.len:=0) | RUN |
| 28 | RUN | DL_MCT_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=DL<br>=><br>DMPMM1_Data_Exchange.cnf(+)(Rem_Add:=S_addr, Diag_Flag:=False, Inp_Data:=DLSDU) | RUN |
| 29 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=DL<br>=><br>DMPMM1_Data_Exchange.cnf(+)(Rem_Add:=S_addr, Diag_Flag:=False, Inp_Data:=DLSDU) | RUN |
| 30 | RUN | DL_MCT_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=DH<br>=><br>DMPMM1_Data_Exchange.cnf(+)(Rem_Add:=S_addr, Diag_Flag:=True, Inp_Data:=DLSDU) | RUN |
| 31 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=DH<br>=><br>DMPMM1_Data_Exchange.cnf(+)(Rem_Add:=S_addr, Diag_Flag:=True, Inp_Data:=DLSDU) | RUN |
| 32 | RUN | DL_MCT_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=DS/UE/RS/NA<br>=><br>DMPMM1_Data_Exchange.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 33 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=DS/UE/RS/NA<br>=><br>DMPMM1_Data_Exchange.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 34 | RUN | DL_MCT_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=RR/RDL/RDH<br>=><br>DMPMM1_Data_Exchange.cnf(-)(Rem_Add:=S_addr, Status:=RR) | RUN |
| 35 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=NIL && D_SAP_index=NIL && DL_status=RR/RDL/RDH<br>=><br>DMPMM1_Data_Exchange.cnf(-)(Rem_Add:=S_addr, Status:=RR) | RUN |
| 36 | RUN | DMPMM1_Slave_Diag.req(Rem_Add)<br><br>=><br><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=60, S_addr:=Rem_Add,DLSDU:=ZERO-PDU, Service_class:=High) | RUN |
| 37 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DL && DLSDU.len >= 6<br>&& DLSDU.len <= Act_Max_Diag_Len<br>&& (Diagnosis-RES-PDU.Diag_Master_Add=255 \|\| Diagnosis-RES-PDU.Diag_Master_Add=Loc_Station_Address)<br>=><br>DMPMM1_Slave_Diag.cnf(+)(Rem_Add:=S_addr, Diag_Data:=DLSDU) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 38 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DL && DLSDU.len > Act_Max_Diag_Len<br>&& (Diagnosis-RES-PDU.Diag_Master_Add<>255 && Diagnosis-RES-PDU.Diag_Master_Add<>Loc_Station_Address)<br>=><br>cut Diagnosis-RES-PDU until Act_Max_Diag_Len<br>Diagnosis-RES-PDU.Station_status_3.Ext_Diag_Overflow:=1<br>Diagnosis-RES-PDU.Station_status_1.Master_Lock := 1<br>DMPMM1_Slave_Diag.cnf(+)(Rem_Add:=S_addr, Diag_Data:=Diagnosis-RES-PDU) | RUN |
| 39 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DL && DLSDU.len > Act_Max_Diag_Len<br>&& (Diagnosis-RES-PDU.Diag_Master_Add=255 \|\| Diagnosis-RES-PDU.Diag_Master_Add=Loc_Station_Address)<br>=><br>cut Diagnosis-RES-PDU until Act_Max_Diag_Len<br>Diagnosis-RES-PDU.Station_status_3.Ext_Diag_Overflow:=1<br>DMPMM1_Slave_Diag.cnf(+)(Rem_Add:=S_addr, Diag_Data:=Diagnosis-RES-PDU) | RUN |
| 40 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DL && DLSDU.len >= 6<br>&& DLSDU.len <= Act_Max_Diag_Len<br>&& (Diagnosis-RES-PDU.Diag_Master_Add<>255 && Diagnosis-RES-PDU.Diag_Master_Add<>Loc_Station_Address)<br>=><br>Diagnosis-RES-PDU.Station_status_1.Master_Lock := 1<br>DMPMM1_Slave_Diag.cnf(+)(Rem_Add:=S_addr, Diag_Data:=Diagnosis-RES-PDU) | RUN |
| 41 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DL && DLSDU.len < 6<br>=><br>DMPMM1_Slave_Diag.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 42 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DH/RDL/RDH<br>=><br>DMPMM1_Slave_Diag.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 43 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DS/UE/RS/NA/NR/RR<br>=><br>DMPMM1_Slave_Diag.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 44 | RUN | DMPMM1_Global_Control.req(Rem_Add, Control_Command, Group_Select)<br>/Isochron Mode Supp = FALSE \|\| Isochron Mode = Not Synchronized<br>\|\| Group_Select <> 0 \|\| Control_Command&2=2 && DL IsoM On=FALSE<br>\|\| Control_Command&2=0 && DL IsoM On=TRUE<br>=><br>DL_DATA.req(S_SAP_index:=62, D_SAP_index:=58, D_addr:=Rem_Add,DLSDU:=Global_Control-REQ-PDU,Service_class:=High) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 45 | RUN | DMPMM1_Global_Control.req(Rem_Add, Control_Command, Group_Select)<br>/Isochron Mode Supp = TRUE && Isochron Mode <>Not Synchronized &&<br>Group_Select=0 && Control_Command&2=0 && DL IsoM On=FALSE<br>=><br>DL IsoM On:=TRUE<br>Control_Command:=0<br>if(IsoM Sync)<br>Control_Command.Sync:=1<br>endif<br>if(IsoM Freeze)<br>Control_command.Freeze:=1<br>endif<br>DLSDU:=Global_Control-REQ-PDU(Control_Command, Group_Select:=Group_8)<br>DL_DATA.req(S_SAP_index:=62, D_SAP_index:=58, D_addr:=Rem_Add,<br>DLSDU:=Global_Control-REQ-PDU,<br>Service_class:=High)<br>DLM_SET_VALUE.req( Variable_name1:="SYNCHT",<br>Variable_name2:="maxTsh",<br>Variable_name3:="Tct",<br>Desired_Value1:=DLSDU,<br>Desired_Value2:=maxTsh,<br>Desired_Value3:=Tct) | RUN |
| 46 | RUN | DMPMM1_Global_Control.req(Rem_Add, Control_Command, Group_Select)<br>/Isochron Mode Supp = TRUE && Isochron Mode <>Not Synchronized &&<br>Group_Select=0 && Control_Command&2=2 && DL IsoM On=TRUE<br>=><br>DL IsoM On:=FALSE<br>DLSDU:=Global_Control-REQ-PDU(Control_Command:= 2, Group_Select:=Group_8)<br>DL_DATA.req(S_SAP_index:=62, D_SAP_index:=58, D_addr:=Rem_Add,<br>DLSDU:=Global_Control-REQ-PDU,<br>Service_class:=High)<br>DLM_SET_VALUE.req( Variable_name1:="SYNCHT",<br>Variable_name2:="maxTsh",<br>Variable_name3:="Tct",<br>Desired_Value1:=DLSDU,<br>Desired_Value2:=maxTsh,<br>Desired_Value3:=Tct) | RUN |
| 47 | RUN | DL_DATA.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class,<br>DL_status)<br>/S_SAP_index=62 && D_SAP_index=58 && DL_status=OK/DS<br>=><br>DMPMM1_Global_Control.cnf(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 48 | RUN | DMPMM1_Chk_Cfg.req(Rem_Add, Cfg_Data)<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=62, S_addr:=Rem_Add,<br>DLSDU:=Cfg_Data, Service_class:=High) | RUN |
| 49 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class,<br>DL_Status)<br>/S_SAP_index=62 && D_SAP_index=62 && DL_status=NR<br>=><br>DMPMM1_Chk_Cfg.cnf(+)(Rem_Add:=S_addr) | RUN |
| 50 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class,<br>DL_Status)<br>/S_SAP_index=62 && D_SAP_index=62 && DL_status=DS/NA/RS/RR/UE<br>=><br>DMPMM1_Chk_Cfg.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 51 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class,<br>DL_Status)<br>/S_SAP_index=62 && D_SAP_index=62 && DL_status=RDL/RDH/DL/DH<br>=><br>DMPMM1_Chk_Cfg.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 52 | RUN | DMPMM1_Set_Prm.req(Rem_Add, Prm_Data)<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=61, S_addr:=Rem_Add,<br>DLSDU:=Prm_Data, Service_class:=High) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 53 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=61 && DL_status=NR<br>=><br>DMPMM1_Set_Prm.cnf(+)(Rem_Add:=S_addr) | RUN |
| 54 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=61 && DL_status=DS/NA/RS/RR/UE<br>=><br>DMPMM1_Set_Prm.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 55 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=61 && DL_status=RDL/RDH/DL/DH<br>=><br>DMPMM1_Set_Prm.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 56 | RUN | DMPMM1_CS_TIME_EVENT.req<br>=><br>D_addr=127, D_SAP_index=CS, S_SAP_index=CS<br>DL_CS_TIME_EVENT.req(D_addr, D_SAP_index, S_SAP_index) | RUN |
| 57 | RUN | DL_CS_TIME_EVENT.cnf (D_addr, D_SAP_index, S_SAP_index, Send_delay_time, DL_status)<br>=><br>Send_Delay_Time=Send_delay_time, DL_Status=DL_status<br>DMPMM1_CS_TIME_EVENT.cnf (Send_Delay_Time, DL_Status) | RUN |
| 58 | RUN | DMPMM1_CS_CLOCK_VALUE.req (Clock Value Time Event, Clock Value previous TE, Clock Value Status)<br>=><br>D_addr=127, D_SAP_index=CS, S_SAP_index=CS, DLSDU=Clock Value Time Event, Clock Value previous TE, Clock Value Status<br>DL_CS_CLOCK_VALUE.req (D_addr, D_SAP_index, S_SAP_index, DLSDU) | RUN |
| 59 | RUN | DL_CS_CLOCK_VALUE.cnf (D_addr, D_SAP_index, S_SAP_index, DL_status)<br>=><br>DL_Status=DL_status<br>DMPMM1_CS_CLOCK_VALUE.cnf (DL_Status) | RUN |
| 60 | RUN | DL_CS_CLOCK_VALUE.ind (D_addr, D_SAP_index, S_addr, S_SAP_index, DLSDU, Receive_delay_time, CS_Status)<br>=><br>Time_Master_Addr=S_addr, CS_list=DLSDU, CS_status=CS_Status, Receive_Delay_Time=Receive_delay_time)<br>DMPMM1_CS_CLOCK_VALUE.ind (Time_Master_Addr, CS_list, CS_status, Receive_Delay_Time) | RUN |
| 61 | RUN | DMPMM1_Set_ExtPrm.req(Rem_Add, Prm_Data)<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=53, S_addr:=Rem_Add,DLSDU:=Prm_Data, Service_class:=High) | RUN |
| 62 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=53 && DL_status=NR<br>=><br>DMPMM1_Set_ExtPrm.cnf(+)(Rem_Add:=S_addr) | RUN |
| 63 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=53 && DL_status=DS/NA/RS/RR/UE<br>=><br>DMPMM1_Set_ExtPrm.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 64 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_Status)<br>/S_SAP_index=62 && D_SAP_index=53 && DL_status=RDL/RDH/DL/DH<br>=><br>DMPMM1_Set_ExtPrm.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 65 | RUN | DLM_SET_VALUE.cnf( DLM_status(1-3))<br>/DLM_status(1-3)=OK<br>=><br>no action | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | Current State | Event /Condition =>Action | Next State |
|---|---|---|---|
| 66 | RUN | DLM_SET_VALUE.cnf( DLM_status(1))<br>/DLM_status(1)=OK<br>=><br>no action | RUN |
| 67 | any state | DMPMM1_Reset.req<br>=><br>DLM_RESET.req(DLMSAP_0) | PON |
| 68 | any state | DLM_RESET.cnf( DLM_status)<br>/DLM_status=OK<br>=><br>DMPMM1_Reset.cnf | PON |
| 69 | any state | inadmissible or unknown DL primitive<br>=><br>DMPMM1_Fault.ind | PON |

### 10.2.4 Functions

Table 131 contains the functions used by the DMPMM1, their arguments and their descriptions.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 131 – Functions used by the DMPMM1**

| Function name | Description |
|---|---|
| ActivateSAPFrom CRList() | This function checks the CRL DP-master (Class 1) if there are CRL entries which SSAP has not been activated with the DLAP-ACTIVATE.req service primitive yet.<br>A) If there exist one of the following CRL entries that has not been activated yet:<br>Entries with SSAP=NIL or SSAP=62 or SSAP=51 or SSAP=54<br>than it returns TRUE and sets the local context according to the current CREP.<br>B) Otherwise it returns FALSE. |
| BuildServiceListFrom CR() | This function builds the Service_list parameter for DLSAP-ACTIVATE.req primitive from the current CRL context.<br>A) If SSAP = NIL(MS0) it returns<br>Service_list.Service_activate[1] = SRD<br>Service_list.Role_in_Service[1] = Initiator<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=244<br>B) If SSAP = 62(MS0) it returns<br>Service_list.Service_activate[1] = SRD<br>Service_list.Role_in_Service[1] = Initiator<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=0<br>Service_list.Service_activate[2] = SDN<br>Service_list.Role_in_Service[2] = Initiator<br>Service_list.DLSDU_length_list[2].Max_DLSDU_length_req_low=0<br>Service_list.DLSDU_length_list[2].Max_DLSDU_length_req_high=2<br>Service_list.DLSDU_length_list[2].Max_DLSDU_length_ind/cnf_low=0<br>Service_list.DLSDU_length_list[2].Max_DLSDU_length_ind/cnf_high=0<br>C) If SSAP = 51(MS1) it returns<br>Service_list.Service_activate[1] = SRD<br>Service_list.Role_in_Service[1] = Initiator<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=0<br>D) If SSAP = 54(MM) it returns<br>Service_list.Service_activate[1] = SDN<br>Service_list.Role_in_Service[1] = Receiver<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=4<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=0 |
| BuildLengthFromCR() | This function builds the Service_list parameter for DLSAP-ACTIVATE-RESPONDER.req primitive from the current CRL context.<br>DLSDU_length_list.Max_DLSDU_length_req_low = 244<br>DLSDU_length_list.Max_DLSDU_length_req_high = 0<br>DLSDU_length_list.Max_DLSDU_length_ind_low =244<br>DLSDU_length_list.Max_DLSDU_length_ind_high = 0 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

## 10.3 DMPMM2

### 10.3.1 Primitive definitions

#### 10.3.1.1 Primitives exchanged between FSPMM2 and DMPMM2

Table 132 shows the service primitives including their associated parameters issued by the FSPMM2 and received by the DMPMM2.

**Table 132 – Primitives issued by FSPMM2 to DMPMM2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit DLL.req | FSPMM2 | Bus Para | Refer to FAL Service Definition in IEC 61158-5-3 and in IEC 61158-3-3 |
| Reset.req | FSPMM2 | (none) | |
| Read Slave Diag.req | FSPMM2 | Rem Add | |
| Read Output.req | FSPMM2 | Rem Add | |
| Read Input.req | FSPMM2 | Rem Add | |
| Get Cfg.req | FSPMM2 | Rem Add | |
| Set Slave Add.req | FSPMM2 | Rem Add,<br>New Slave Add,<br>Ident Number,<br>No Add Chg,<br>Rem Slave Data | |

Table 133 shows the service primitives including their associated parameters issued by the DMPMM2 and received by the FSPMM2.

**Table 133 – Primitives issued by DMPMM2 to FSPMM2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| MInit DLL.cnf | DMPMM2 | (non) | Refer to FAL Service Definition in IEC 61158-5-3 |
| Reset.cnf | DMPMM2 | (none) | |
| Read Slave Diag.cnf(+) | DMPMM2 | Rem Add,<br>Diag Data | |
| Read Slave Diag.cnf(-) | DMPMM2 | Rem Add,<br>Status | |
| Get Cfg.cnf(+) | DMPMM2 | Rem Add,<br>Cfg Data | |
| Get Cfg.cnf(-) | DMPMM2 | Rem Add<br>Status | |
| Read Output.cnf(+) | DMPMM2 | Rem Add,<br>Output Data | |
| Read Output.cnf(-) | DMPMM2 | Rem Add,<br>Status | |
| Read Input.cnf(+) | DMPMM2 | Rem Add,<br>Input Data | |
| Read Input.cnf(-) | DMPMM2 | Rem Add,<br>Status | |
| Set Slave Add.cnf(+) | DMPMM2 | Rem Add | |
| Set Slave Add.cnf(-) | DMPMM2 | Rem Add,<br>Status | |
| Event.ind | DMPMM2 | Event,<br>Add Info | |
| Fault.ind | DMPMM2 | Rem Add | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 10.3.1.2 Primitives exchanged between DMPMM2 and MSAC2M

Table 134 shows the service primitives including their associated parameters issued by the MSAC2M and received by the DMPMM2.

**Table 134 – Primitives issued by MSAC2M to DMPMM2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DATA_REPLY.req | MSAC2M | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class | Send Service |

Table 135 shows the service primitives including their associated parameters issued by the DMPMM2 and received by the MSAC2M.

**Table 135 – Primitives issued by DMPMM2 to MSAC2M**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DATA_REPLY.cnf | DMPMM2 | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class,<br>L_status | Send Service executed |

#### 10.3.1.3 Primitives exchanged between DMPMM2 and MMAC2

Table 136 shows the service primitives including their associated parameters issued by the MMAC2 and received by the DMPMM2.

**Table 136 – Primitives issued by MMAC2 to DMPMM2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DATA_REPLY.req | MMAC2 | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class | Send Service |
| DATA.req | MMAC2 | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class | Send Service |

Table 137 shows the service primitives including their associated parameters issued by the DMPMM2 and received by the MMAC2.

**Table 137 – Primitives issued by DMPMM2 to MMAC2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DATA_REPLY.cnf | DMPMM2 | SSAP,<br>DSAP,<br>Rem_Add,<br>L_sdu,<br>Serv_class,<br>L_status | Send Service executed |
| DATA.cnf | DMPMM2 | SSAP,<br>DSAP,<br>Rem_Add,<br>Serv_class,<br>L_status | Send Service executed |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

#### 10.3.1.4 Primitives exchanged between DMPMM2 and DL

Table 138 shows the service primitives including their associated parameters issued by the DMPMM2 and received by the DL.

**Table 138 – Primitives issued by DMPMM2 to DL**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DL-DATA.req | DMPMM2 | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DLSDU | Refer to DL Service Definition in IEC 61158-3-3 |
| DL-DATA-REPLY.req | DMPMM2 | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DLSDU | |
| DLM-RESET.req | DMPMM2 | (none) | |
| DLM-SET-VALUE.req | DMPMM2 | Variable_name(1 to n),<br>Desired_value (1 to n) | |
| DLSAP-ACTIVATE.req | DMPMM2 | S_SAP_index,<br>Access,<br>Service_list | |
| DLSAP-DEACTIVATE.req | DMPMM2 | S_SAP_index | |

Table 139 shows the service primitives including their associated parameters issued by the DL and received by the DMPMM2.

**Table 139 – Primitives issued by DL to DMPMM2**

| Primitive name | Source | Associated parameters | Functions |
|---|---|---|---|
| DL-DATA.cnf | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DL_status | Refer to DL Service Definition in IEC 61158-3-3 |
| DL-DATA-REPLY.cnf | DL | Service_class,<br>D_addr,<br>D_SAP_index,<br>S_SAP_index,<br>DLSDU,<br>DL_status | |
| DLM-RESET.cnf | DL | DLM_Status | |
| DLM-SET-VALUE.cnf | DL | Current_value (1 to n),<br>DLM_status | |
| DLM-EVENT.ind | DL | Event/Fault | |
| DLSAP-ACTIVATE.cnf | DL | S_SAP_index,<br>DLM_status | |
| DLSAP-DEACTIVATE.cnf | DL | S_SAP_index,<br>DLM_status | |

#### 10.3.1.5 Parameters of FSPMM2 primitives

The parameters used with the primitives exchanged between DMPMM2 and FSPMM2 are described in FAL Service Definition in IEC 61158-5-3. Additional parameters are described in Table 140.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 140 – Parameters used with primitives exchanged with DMPMM2**

| Parameter name | Description |
|---|---|
| SSAP | Local Identifier for Service Access Point |
| DSAP | Identifier for the remote Service Access Point |
| Serv_class | Indicates priority of the service |
| L_status | Status of the service execution |
| L_sdu | Defines the DP PDU to be transmitted |
| Rem_Add | DL Address of the remote station |

### 10.3.2 State machine description

The DMPMM2 makes the connection between the MS0,MS2,MM1/2 specific parts and Layer 2. Services are mapped by the DMPMM2 to the DL and management of Layer 2. The DMPMM2 delivers the necessary Layer 2 parameters of the function call (SSAP, DSAP, Serv_class, ...) for Layer 2, receives the confirmations and indications from Layer 2 and passes them to the MS0,MS2-Handler.

### 10.3.3 DMPMM2 state table

Table 141 contains the complete description of the DMPMM2 state machine.

**Table 141 – DMPMM2 state Table**

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 1 | PON | DMPMM2_Minit_DLL.req(Bus_Para)<br>=><br>DLM_SET_VALUE.req(<br>Variable_name1:= “TS “,<br>Variable_name2:= “Data_rate “,<br>Variable_name3:= “TSL “,<br>Variable_name4:= “min TSDR “,<br>Variable_name5:= “max TSDR “,<br>Variable_name6:= “TQUI “,<br>Variable_name7:= “TSET “,<br>Variable_name8:= “TTR “,<br>Variable_name9:= “G “,<br>Variable_name10:= “HSA “,<br>Variable_name11:= “max_retry_limit “,<br>Variable_name12:= “in_ring_desired”,<br>Desired_Value1:= Bus_Para.TS<br>Desired_Value2:= Bus_Para.Data_rate,<br>Desired_Value3:= Bus_Para.TSL,<br>Desired_Value4:= Bus_Para.min TSDR,<br>Desired_Value5:= Bus_Para.max TSDR,<br>Desired_Value6:= Bus_Para.TQUI,<br>Desired_Value7:= Bus_Para.TSET,<br>Desired_Value8:= Bus_Para.TTR,<br>Desired_Value9:= Bus_Para.G,<br>Desired_Value10:= Bus_Para.HSA,<br>Desired_Value11:= Bus_Para.max_retry_limit,<br>Desired_Value12:= TRUE) | SET-VALUE |
| 2 | PON | DLM_RESET.cnf( DLM_status)<br>/DLM_status=OK<br>=><br>DMPMM2_Reset.cnf | PON |
| 3 | SET-VALUE | DLM_SET_VALUE.cnf( DLM_status(1-11))<br>/DLM_status(1-11)=OK<br>=><br>no action | SAP-CHECK |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 4 | SAP-CHECK | spontaneous<br>/ActivateSAPFromCRList()=TRUE<br>=><br>DLSAP_ACTIVATE.req( S_SAP_index:=SSAP, Access:=All, Service_list:=BuildServiceListFromCR()) | SAP-ACT |
| 5 | SAP-CHECK | spontaneous<br>/ActivateSAPFromCRList()=FALSE<br>=><br>DMPMM2_Minit_DLL.cnf | RUN |
| 6 | SAP-ACT | DLSAP_ACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status=OK<br>=><br>no action | SAP-CHECK |
| 7 | RUN | DLM_EVENT.ind(Event/Fault, Add_info)<br>=><br>DMPMM2_Event.ind(Event:=Event/Fault, Add_Info:=Add_info) | RUN |
| 8 | RUN | DMPMM2_Read_Slave_Diag.req(Rem_Add)<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=60, S_addr:=Rem_Add, DLSDU.len:=0, Service_class:=High) | RUN |
| 9 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DL && DLSDU.len >= 6<br>=><br>DMPMM2_Read_Slave_Diag.cnf(+)(Rem_Add:=S_addr, Diag_Data:=DLSDU) | RUN |
| 10 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DL && DLSDU < 6<br>=><br>DMPMM2_Read_Slave_Diag.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 11 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DH/RDL/RDH<br>=><br>DMPMM2_Read_Slave_Diag.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 12 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=60 && DL_status=DS/UE/RS/NA/NR/RR<br>=><br>DMPMM2_Read_Slave_Diag.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 13 | RUN | DMPMM2_Get_Cfg.req(Rem_Add)<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=59, S_addr:=Rem_Add, DLSDU.len:=0, Service_class:=High) | RUN |
| 14 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=59 && DL_status=DL<br>=><br>DMPMM2_Get_Cfg.cnf(+)(Rem_Add:=S_addr, Cfg_Data:=DLSDU) | RUN |
| 15 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=59 && DL_status=DS/UE/RS/NA/NR/RR<br>=><br>DMPMM2_Get_Cfg.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 16 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=59 && DL_status=DH/RDL/RDH<br>=><br>DMPMM2_Get_Cfg.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 17 | RUN | DMPMM2_Read_Input.req(Rem_Add)<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=56, S_addr:=Rem_Add, DLSDU.len:=0, Service_class:=High) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 18 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=56 && DL_status=DL<br>=><br>DMPMM2_Read_Input.cnf(+)(Rem_Add:=S_addr, Inp_Data:=DLSDU) | RUN |
| 19 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=56 && DL_status=DS/UE/RS/NA/NR/RR<br>=><br>DMPMM2_Read_Input.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 20 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=56 && DL_status=DH/RDL/RDH<br>=><br>DMPMM2_Read_Input.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 21 | RUN | DMPMM2_Read_Output.req(Rem_Add)<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=57, S_addr:=Rem_Add, DLSDU.len:=0, Service_class:=High) | RUN |
| 22 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=57 && DL_status=DL<br>=><br>DMPMM2_Read_Output.cnf(+)(Rem_Add:=S_addr, Outp_Data:=DLSDU) | RUN |
| 23 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=57 && DL_status=DS/UE/RS/NA/NR/RR<br>=><br>DMPMM2_Read_Output.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 24 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=57 && DL_status=DH/RDL/RDH<br>=><br>DMPMM2_Read_Output.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 25 | RUN | DMPMM2_Set_Slave_Add.req(Rem_Add, New_Slave_Add, Ident_Number, No_Add_Chg, Rem_Slave_Data)<br>=><br>DL_DATA_REPLY.req(S_SAP_index:=62, D_SAP_index:=55, S_addr:=Rem_Add, DLSDU:=Set_Slave_Add-REQ-PDU, Service_class:=High) | RUN |
| 26 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=55 && DL_status=NR<br>=><br>DMPMM2_Set_Slave_Add.cnf(+)(Rem_Add:=S_addr) | RUN |
| 27 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=55 && DL_status=DS/UE/RR/RS/NA<br>=><br>DMPMM2_Set_Slave_Add.cnf(-)(Rem_Add:=S_addr, Status:=DL_status) | RUN |
| 28 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index=62 && D_SAP_index=55 && DL_status=RDL/RDH/DL/DH<br>=><br>DMPMM2_Set_Slave_Add.cnf(-)(Rem_Add:=S_addr, Status:=RE) | RUN |
| 29 | RUN | DMPMM2_DATA_REPLY.req (SSAP, DSAP, Rem_Add, L_sdu, Serv_class )<br>=><br>DL_DATA_REPLY.req (S_SAP_index:=SSAP, D_SAP_index:=DSAP, S_addr:=Rem_Add, DLSDU:=L_sdu, Service_class:=Serv_class ) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | Current state | Event / condition => action | Next state |
|---|---|---|---|
| 30 | RUN | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/S_SAP_index<>62 && DL_status<>LS/IV/LR<br>=><br>DMPMM2_DATA_REPLY.cnf (SSAP:=S_SAP_index, DSAP:=D_SAP_index, Rem_Add:=S_addr, L_sdu:=DLSDU, Serv_class:=Service_class, L_status:=DL_status ) | RUN |
| 31 | RUN | DMPMM2_DATA.req (SSAP, DSAP, Rem_Add, L_sdu, Serv_class )<br>=><br>DL_DATA.req (S_SAP_index:=SSAP, D_SAP_index:=DSAP, S_addr:=Rem_Add, DLSDU:=L_sdu, Service_class:=Serv_class ) | RUN |
| 32 | RUN | DL_DATA.cnf(S_SAP_index, D_SAP_index, S_addr, Service_class, DL_status)<br>/DL_status<>LS/IV/LR<br>=><br>DMPMM2_DATA.cnf (SSAP:=S_SAP_index, DSAP:=D_SAP_index, Rem_Add:=S_addr, Serv_class:=Service_class, L_status:=DL_status ) | RUN |
| 33 | any state | DLSAP_ACTIVATE.cnf( S_SAP_index, DLM_status)<br>/DLM_status<>OK<br>=><br>DMPMM2_Fault.ind | PON |
| 34 | any state | DLM_SET_VALUE.cnf( DLM_status(1-11))<br>/NOT (DLM_status(1-11)=OK)<br>=><br>DMPMM2_Fault.ind | PON |
| 35 | any state | DL_DATA_REPLY.cnf(S_SAP_index, D_SAP_index, S_addr, DLSDU, Service_class, DL_status)<br>/DL_status=LS/IV/LR<br>=><br>DMPMM2_Fault.ind | RUN |
| 36 | any state | inadmissible or unknown DL primitive<br>=><br>DMPMM2_Fault.ind | RUN |
| 37 | any state | DL_DATA.cnf(S_SAP_index, D_SAP_index, S_addr, Service_class, DL_status)<br>/DL_status=LS/IV/LR<br>=><br>DMPMM2_Fault.ind | PON |
| 38 | any state | DMPMM2_Reset.req<br>=><br>DLM_RESET.req | PON |

### 10.3.4 Functions

Table 142 contains the functions used by the DMPMM2, their arguments and their descriptions.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Table 142 – Functions used by DMPMM2**

| Function name | Description |
|---|---|
| ActivateSAPFromCRList() | This function checks the CRL DP-master (Class 2) if there are CRL entries which SSAP has not been activated with the DLM-SAP-ACTIVATE.req service primitive yet.<br>A) If there exist one of the following CRL entries that has not been activated yet:<br>Entries with SSAP=62 or SSAP=50 or SSAP=54<br>than it returns TRUE and sets the local context according to the current CREP.<br>B) Otherwise it returns FALSE. |
| BuildServiceListFromCR() | This function builds the Service_list parameter for DLSAP-ACTIVATE.req primitive from the current CRL context.<br>A) If SSAP = 62(MS0) it returns<br>Service_list.Service_activate[1] = SRD<br>Service_list.Role_in_Service[1] = Initiator<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=0<br>B) If SSAP = 50(MS2) it returns<br>Service_list.Service_activate[1] = SRD<br>Service_list.Role_in_Service[1] = Initiator<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=0<br>C) If SSAP = 54(MM) it returns<br>Service_list.Service_activate[1] = SRD<br>Service_list.Role_in_Service[1] = Initiator<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_req_high=0<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_low=244<br>Service_list.DLSDU_length_list[1].Max_DLSDU_length_ind/cnf_high=0<br>Service_list.Service_activate[2] = SDN<br>Service_list.Role_in_Service[2] = Initiator<br>Service_list.DLSDU_length_list[2].Max_DLSDU_length_req_low=2<br>Service_list.DLSDU_length_list[2].Max_DLSDU_length_req_high=0<br>Service_list.DLSDU_length_list[2].Max_DLSDU_length_ind/cnf_low=0<br>Service_list.DLSDU_length_list[2].Max_DLSDU_length_ind/cnf_high=0 |

## 11 Parameters for a DP-slave

Table 143 contains limitations for datarate-depending AL timing parameters of Slaves.

**Table 143 – Bus parameter/reaction times for a DP-slave**

| Datarate (kbit/s) | ≤ 187,5 | 500 | 1 500 | 3 000 | 6 000 | 12 000 |
|---|---|---|---|---|---|---|
| Min_Slave_Interval (ms) | ≤ 20 | ≤ 6 | ≤ 2 | ≤ 1,5 | ≤ 1 | ≤ 0,6 |
| Send_Timeout (sec) (1) | 4 | 2 | 1 | 1 | 1 | 1 |
| MS1_Timeout (sec) (2) | 40 | 20 | 10 | 10 | 10 | 10 |

NOTE 1 The parameter Send_Timeout represents a performance feature of a DP-slave implementation. Each DP-slave implementation has to ensure that the parameter Send_Timeout reaches the smallest value that is possible.

NOTE 2 The parameter MS1_Timeout represents a performance feature of a DP-slave. Each DP-slave implementation has to ensure that the parameter MS1_Timeout reaches the smallest value that is possible.

Another performance feature for a DP-slave implementation is the time Min_Slave_Interval. Each DP-slave implementation has to ensure that the Min_Slave_Interval reaches the smallest possible value. This means, that in a DP-System with more than ten stations the Min_Slave_Interval shall not be the dominant factor for the cycle time.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

## Bibliography

IEC/TR 61158-1:2010[2], *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

2 To be published

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

# SOMMAIRE

AVANT-PROPOS ..... 384
INTRODUCTION ..... 386
1 Domaine d'application ..... 388
1.1 Généralités ..... 388
1.2 Spécifications ..... 389
1.3 Conformité ..... 389
2 Références normatives ..... 389
3 Termes, définitions, abréviations, symboles et conventions ..... 390
3.1 Termes et définitions référencés ..... 390
3.2 Définitions supplémentaires ..... 391
3.3 Abréviations et symboles ..... 394
3.4 Conventions ..... 396
3.5 Conventions utilisées dans les diagrammes d'états ..... 398
4 Description de la syntaxe de FAL ..... 401
4.1 Syntaxe abstraite des APDU ..... 401
4.2 Types de données ..... 405
5 Syntaxe de transfert ..... 407
5.1 Codage des types de données de base ..... 407
5.2 Section de codage relative aux PDU d'échange de données ..... 410
5.3 Section de codage relative aux PDU d'échange de diagnostic d'esclave ..... 411
5.4 Section de codage relative à la PDU de paramétrisation ..... 421
5.5 Section de codage relative aux configurationPDUs ..... 428
5.6 Section de codage relative aux PDU de commande globale ..... 432
5.7 Section de codage relative aux clock-value-PDU ..... 433
5.8 Section de codage relative à l'identification de fonction et aux erreurs ..... 434
5.9 Section de codage relative à la PDU de diagnostic de maître ..... 439
5.10 Section de codage relative aux PDU upload/download/act para ..... 441
5.11 Section de codage relative au jeu de paramètres de bus ..... 443
5.12 Section de codage relative au jeu de paramètres d'esclave ..... 446
5.13 Section de codage relative aux compteurs statistiques ..... 450
5.14 Section de codage relative à la PDU d'établissement d'adresse esclave ..... 450
5.15 Section de codage relative aux PDU initiate/abort ..... 450
5.16 Section de codage relative aux PDU read/write/data transport ..... 454
5.17 Section de codage relative aux PDU région de charge et invocation de fonction ..... 454
5.18 Exemples des diagnosis-RES-PDU ..... 458
5.19 Exemple de Chk_Cfg-REQ-PDU ..... 460
5.20 Exemples de Chk_Cfg-REQ-PDU avec types de données DPV1 ..... 461
5.21 Exemple de structure de Data_Unit pour Data_Exchange ..... 463
6 Diagrammes d'états de protocole FAL ..... 464
6.1 Structure globale ..... 464
6.2 Affectation des diagrammes d'états à des dispositifs ..... 466
6.3 Vue d'ensemble d'un esclave DP ..... 466
6.4 Vue d'ensemble de maître DP (Classe 1) ..... 468
6.5 Vue d'ensemble de maître DP (Classe 2) ..... 469
6.6 Communication cyclique entre maître DP (Classe 1) et esclave DP ..... 470

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

6.7 Communication acyclique entre maître DP (Classe 2) et maître DP (Classe 1) ..... 471
6.8 Communication acyclique entre maître DP (Classe 1) et esclave DP ..... 473
6.9 Surveillance d'une relation d'applications ..... 475
7 Diagramme d'états AP-Context (contexte d'AP) ..... 481
8 Diagrammes protocolaires de services de la FAL (FSPM) ..... 481
8.1 FSPMS ..... 481
8.2 FSPMM1 ..... 516
8.3 FSPMM2 ..... 552
9 Diagramme d'état de protocol de mapping d'application (ARPM) ..... 570
9.1 MSCY1S ..... 570
9.2 MSAC1S ..... 602
9.3 SSCY1S ..... 615
9.4 MSRM2S ..... 618
9.5 MSAC2S ..... 624
9.6 MSCS1S ..... 641
9.7 MSCY1M ..... 643
9.8 MSAL1M ..... 663
9.9 MSAC1M ..... 672
9.10 MMAC1 ..... 686
9.11 MSCS1M ..... 693
9.12 MSAC2M ..... 698
9.13 MMAC2 ..... 713
10 Diagrammes protocolaires de mise en correspondance de DLL (les DMPM) ..... 720
10.1 DMPMS ..... 720
10.2 DMPMM1 ..... 733
10.3 DMPMM2 ..... 750
11 Paramètres pour un esclave DP ..... 758
Bibliographie ..... 759

Figure 1 – Structure commune des champs spécifiques ..... 397
Figure 2 – Codage du type de données BinaryDate ..... 408
Figure 3 – Codage d'une valeur TimeOfDay ..... 409
Figure 4 – Codage d'une valeur Time Difference ..... 409
Figure 5 – Codage d'une valeur Network Time ..... 410
Figure 6 – Codage d'une valeur Network Time Difference ..... 410
Figure 7 – Exemple de Modul_Status_Array ..... 416
Figure 8 – Exemple d'Ext_Diag_Data dans le cas du format de diagnostic DPV1 avec PDU d'alarme et de statut ..... 459
Figure 9 – Exemple d'Ext_Diag_Data dans le cas du format de diagnostic de base ..... 460
Figure 10 – Exemple de format d'identificateur spécial ..... 461
Figure 11 – Exemple de format d'identificateur spécial avec types de données ..... 461
Figure 12 – Exemple de format d'identificateur spécial avec types de données ..... 462
Figure 13 – Exemple de position vide avec types de données ..... 462
Figure 14 – Exemple de dispositif à plusieurs variables avec des blocs de fonction AI et DO ..... 463
Figure 15 – Identificateurs (ID) ..... 463
Figure 16 – Liste d'identificateurs ..... 463

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Figure 17 – Structure de la Data_Unit pour DLPDU de demande et DLPDU de réponse ..... 464
Figure 18 – Structuration des diagrammes protocolaires et des couches adjacentes dans un esclave DP ..... 467
Figure 19 – Structuration des diagrammes protocolaires et des couches adjacentes dans un maître DP (Classe 1) ..... 468
Figure 20 – Structuration des diagrammes protocolaires et des couches adjacentes dans un maître DP (Classe 2) ..... 469
Figure 21 – Séquence de la communication entre maître DP et esclave DP ..... 471
Figure 22 – Séquence de communication entre maître DP (classe 2) et maître DP (classe 1) ..... 473
Figure 23 – Séquence de communication acyclique entre maître DP (classe 1) et esclave DP ..... 475
Figure 24 – Exemple pour établissement de connexion sur MS2 ..... 478
Figure 25 – Repos côté maître sur MS2 ..... 479
Figure 26 – Repos côté esclave sur MS2 ..... 480
Figure 27 – Exemple pour établissement de connexion sur MS2 (côté serveur) ..... 620
Figure 28 – Structure des entrées RM dans le RM_Registry ..... 621

Tableau 1 – Éléments de description de diagramme d'états ..... 398
Tableau 2 – Description d'éléments de diagramme d'états ..... 398
Tableau 3 – Conventions utilisées dans les diagrammes d'états ..... 399
Tableau 4 – Syntaxe des APDU ..... 401
Tableau 5 – Substitutions ..... 404
Tableau 6 – Plage d'Alarm_Type ..... 413
Tableau 7 – Plage des valeurs de Status_Type ..... 414
Tableau 8 – Alarm_Specifier ..... 414
Tableau 9 – Plage des Modul_Status_Entry (1 à 4) ..... 416
Tableau 10 – Type d'erreur (Error_Type) ..... 418
Tableau 11 – Spécification des bits Lock_Req et Unlock_Req ..... 422
Tableau 12 – Plage de Length_of_Manufacturer_Specific_Data s'il est utilisé dans Chk_Cfg-REQ-PDU ..... 429
Tableau 13 – Plage de Length_of_Manufacturer_Specific_Data s'il est utilisé dans Get_Cfg-RES-PDU ..... 429
Tableau 14 – Valeurs (codes) pour les types de données ..... 431
Tableau 15 – Spécification des bits pour Un-/Sync et Un-/Freeze ..... 432
Tableau 16 – Codage de Function_Code/ Function_Num ..... 435
Tableau 17 – Codage de l'Error_Code/ Function_Num ..... 436
Tableau 18 – Valeurs d'Error_Decode ..... 436
Tableau 19 – Codage d'Error_Code_1 à DPV1 ..... 438
Tableau 20 – Valeurs de MDiag_Identifier ..... 439
Tableau 21 – Valeurs pour Area_Code_UpDownload ..... 441
Tableau 22 – Valeurs pour Area_CodeActBrct ..... 442
Tableau 23 – Valeurs pour Area_CodeAct ..... 443
Tableau 24 – Valeurs pour Data_rate ..... 444
Tableau 25 – Valeurs pour Slave_Type ..... 447

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Tableau 26 – Valeurs pour Alarm_Mode ..... 447
Tableau 27 – Valeurs pour Subnet ..... 453
Tableau 28 – Valeur de code de cause si l'instance est DLL ..... 453
Tableau 29 – Valeur de code de cause si l'instance est MS2 ..... 454
Tableau 30 – Valeurs d'Extended_Function_Num ..... 455
Tableau 31 – Valeurs de FI_State ..... 457
Tableau 32 – Affectation de diagrammes d'états ..... 466
Tableau 33 – Primitives émises par l'AP-Context vers le FSPMS ..... 481
Tableau 34 – Primitives émises par le FSPMS vers l'AP-Context ..... 483
Tableau 35 – Table d'états de FSPMS ..... 490
Tableau 36 – Fonctions utilisées par le FSPMS ..... 515
Tableau 37 – Primitives émises par l'AP-Context vers le FSPMM1 ..... 516
Tableau 38 – Primitives émises par le FSPMM1 vers l'AP-Context ..... 518
Tableau 39 – Table d'états de FSPMM1 ..... 526
Tableau 40 – Fonctions utilisées par le FSPMM1 ..... 552
Tableau 41 – Primitives émises par l'AP-Context vers le FSPMM2 ..... 553
Tableau 42 – Primitives émises par le FSPMM2 vers l'AP-Context ..... 555
Tableau 43 – Table d'états de FSPMM2 ..... 558
Tableau 44 – Fonctions utilisées par le FSPMM2 ..... 570
Tableau 45 – Primitives émises par le FSPMS vers le MSCY1S ..... 570
Tableau 46 – Primitives émises par le MSCY1S vers le FSPMS ..... 571
Tableau 47 – Règles pour la vérification de DPV1_Status_1, DPV1_Status_2 et DPV1_Status_3 ..... 573
Tableau 48 – Table d'états de MSCY1S ..... 579
Tableau 49 – Fonctions utilisées par le MSCY1S ..... 600
Tableau 50 – Primitives émises par FSPMS vers MSAC1S ..... 602
Tableau 51 – Primitives émises par MSAC1S vers FSPMS ..... 603
Tableau 52 – Primitives émises par MSCY1S vers MSAC1S ..... 603
Tableau 53 – Primitives émises par MSAC1S vers MSCY1S ..... 603
Tableau 54 – Paramètre utilisé avec les primitives échangées entre le diagramme MSAC1S et MSCY1S ..... 603
Tableau 55 – Table d'états de MSAC1S ..... 605
Tableau 56 – Fonctions utilisées par le MSAC1S ..... 615
Tableau 57 – Primitives émises par FSPMS vers SSCY1S ..... 615
Tableau 58 – Primitives émises par SSCY1S vers FSPMS ..... 615
Tableau 59 – Table d'états de SSCY1S ..... 617
Tableau 60 – Fonctions utilisées par le SSCY1S ..... 618
Tableau 61 – Primitives émises par FSPMS vers MSRM2S ..... 619
Tableau 62 – Primitives émises par MSRM2S vers FSPMS ..... 619
Tableau 63 – Table d'états de MSRM2S ..... 622
Tableau 64 – Primitives émises par FSPMS vers MSAC2S ..... 625
Tableau 65 – Primitives émises par MSAC2S vers FSPMS ..... 626
Tableau 66 – Primitives émises par MSRM2S vers MSAC2S ..... 626
Tableau 67 – Primitives émises par MSAC2S vers MSRM2S ..... 627

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Tableau 68 – Paramètre utilisé avec les primitives échangées avec MSAC2S ..... 627
Tableau 69 – Table d'états de MSAC2S ..... 630
Tableau 70 – Primitives émises par MSCS1S vers FSPMS ..... 641
Tableau 71 – Table d'états de MSCS1S ..... 643
Tableau 72 – Primitives émises par FSPMM1 vers MSCY1M ..... 644
Tableau 73 – Primitives émises par MSCY1M vers FSPMM1 ..... 645
Tableau 74 – Paramètres utilisés avec les primitives échangées entre FSPMM1 et MSCY1M ..... 645
Tableau 75 – Table d'états de MSCY1M ..... 648
Tableau 76 – Primitives émises par FSPMM1 vers MSAL1M ..... 663
Tableau 77 – Primitives émises par MSAL1M vers FSPMM1 ..... 663
Tableau 78 – Primitives émises par MSCY1M vers MSAL1M ..... 663
Tableau 79 – Primitives émises par MSAL1M vers MSCY1M ..... 664
Tableau 80 – Paramètre utilisé avec les primitives échangées entre MSAL1M et MSCY1M ..... 664
Tableau 81 – Valeurs possibles dans la Alarm_State_Table ..... 665
Tableau 82 – Table d'états de MSAL1M ..... 668
Tableau 83 – Primitives émises par FSPMM1 vers MSAC1M ..... 673
Tableau 84 – Primitives émises par MSAC1M vers FSPMM1 ..... 673
Tableau 85 – Primitives émises par MSAL1M vers MSAC1M ..... 674
Tableau 86 – Primitives émises par MSAC1M vers MSAL1M ..... 674
Tableau 87 – Paramètre utilisé avec les primitives échangées entre MSAL1M et MSCY1M ..... 674
Tableau 88 – Table d'états de MSAC1M ..... 681
Tableau 89 – Primitives émises par FSPMM1 vers MMAC1 ..... 687
Tableau 90 – Primitives émises par MMAC1 vers FSPMM1 ..... 687
Tableau 91 – Table d'états de MMAC1 ..... 689
Tableau 92 – Primitives émises par FSPMM1 vers MSCS1M ..... 694
Tableau 93 – Primitives émises par MSCS1M vers FSPMM1 ..... 694
Tableau 94 – Table d'états de MSCS1M ..... 696
Tableau 95 – Primitives émises par FSPMM2 vers MSAC2M ..... 698
Tableau 96 – Primitives émises par MSAC2M vers FSPMM2 ..... 699
Tableau 97 – Paramètres utilisés avec les primitives échangées avec MSAC2M ..... 699
Tableau 98 – Table d'états de MSAC2M ..... 703
Tableau 99 – Primitives émises par FSPMM2 vers MMAC2 ..... 713
Tableau 100 – Primitives émises par MMAC2 vers FSPMM2 ..... 714
Tableau 101 – Paramètres utilisés avec les primitives échangées avec MMAC2 ..... 714
Tableau 102 – Table d'états de MMAC2 ..... 715
Tableau 103 – Primitives émises par FSPMS vers DMPMS ..... 720
Tableau 104 – Primitives émises par DMPMS vers FSPMS ..... 720
Tableau 105 – Primitives émises par MSCY1S vers DMPMS ..... 721
Tableau 106 – Primitives émises par DMPMS vers MSCY1S ..... 721
Tableau 107 – Primitives émises par DMPMS vers SSCY1S ..... 722
Tableau 108 – Primitives émises par MSAC1S, MSRM2S, MSAC2S vers DMPMS ..... 722
Tableau 109 – Primitives émises par DMPMS vers MSAC1S, MSRM2S, MSAC2S ..... 722
Tableau 110 – Primitives émises par DMPMS vers MSCS1S ..... 723

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Tableau 111 – Primitives émises par DMPMS vers DL ..... 723
Tableau 112 – Primitives émises par DL vers DMPMS ..... 724
Tableau 113 – Paramètres utilisés avec les primitives échangées avec DMPMS ..... 725
Tableau 114 – Table d'états de DMPMS ..... 726
Tableau 115 – Fonctions utilisées par le DMPMS ..... 732
Tableau 116 – Primitives émises par FSPMM1 vers DMPMM1 ..... 733
Tableau 117 – Primitives émises par DMPMM1 vers FSPMM1 ..... 734
Tableau 118 – Primitives émises par MSCY1M vers DMPMM1 ..... 734
Tableau 119 – Primitives émises par DMPMM1 vers MSCY1M ..... 735
Tableau 120 – Primitives émises par MSAL1M, MSAC1M vers DMPMM1 ..... 735
Tableau 121 – Primitives émises par DMPMM1 vers MSAL1M, MSAC1M ..... 735
Tableau 122 – Primitives émises par MMAC1 vers DMPMM1 ..... 736
Tableau 123 – Primitives émises par DMPMM1 vers MMAC1 ..... 736
Tableau 124 – Primitives émises par MSCS1M vers DMPMM1 ..... 736
Tableau 125 – Primitives émises par DMPMM1 vers MSCS1M ..... 737
Tableau 126 – Primitives émises par DMPMM1 vers DL ..... 737
Tableau 127 – Primitives émises par DL vers DMPMM1 ..... 738
Tableau 128 – Paramètres utilisés avec les primitives échangées avec DMPMM1 ..... 739
Tableau 129 – Possible valeurs de statut ..... 740
Tableau 130 – Table d'états de DMPMM1 ..... 742
Tableau 131 – Fonctions utilisées par le DMPMM1 ..... 750
Tableau 132 – Primitives émises par FSPMM2 vers DMPMM2 ..... 751
Tableau 133 – Primitives émises par DMPMM2 vers FSPMM2 ..... 751
Tableau 134 – Primitives émises par MSAC2M vers DMPMM2 ..... 752
Tableau 135 – Primitives émises par DMPMM2 vers MSAC2M ..... 752
Tableau 136 – Primitives émises par MMAC2 vers DMPMM2 ..... 752
Tableau 137 – Primitives émises par DMPMM2 vers MMAC2 ..... 752
Tableau 138 – Primitives émises par DMPMM2 vers DL ..... 753
Tableau 139 – Primitives émises par DL vers DMPMM2 ..... 753
Tableau 140 – Paramètres utilisés avec les primitives échangées avec DMPMM2 ..... 754
Tableau 141 – Table d'états de DMPMM2 ..... 754
Tableau 142 – Fonctions utilisées par DMPMM2 ..... 758
Tableau 143 – Paramètre de bus / temps de réaction pour un esclave DP ..... 758

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

# RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

## Partie 6-3: Spécification de protocole de la couche application – Eléments de Type 3

## AVANT-PROPOS

1) La CEI (Commission Électrotechnique Internationale) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI - entre autres activités - publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
2) Les décisions ou accords officiels de la CEI concernant des questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux intéressés sont représentés dans chaque comité d'études.
3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toute divergence entre toute Publication de la CEI et toute publication nationale ou régionale correspondante doit être indiquée en termes clairs dans cette dernière.
5) La CEI elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de la CEI. La CEI n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.

NOTE 1 L'utilisation de certains des types de protocoles associés est limitée par les détenteurs de leurs droits de propriété intellectuelle. Dans tous les cas, l'engagement à un abandon limité des droits de propriété intellectuelle pris par les détenteurs de ces droits permet d'utiliser un type particulier de protocole de couche liaison de données avec des protocoles de couche physique et de couche application dans des combinaisons de types telles que spécifiées de façon explicite dans les parties profil. L'utilisation des divers types de protocoles dans d'autres combinaisons peut exiger la permission donnée par les détenteurs respectifs de leurs droits de propriété intellectuelle.

La Norme internationale CEI 61158-6-3 a été établie par le sous-comité 65C: Réseaux de communication industriels, du comité d'études 65 de la CEI: Mesure, commande et automation dans les processus industriels.

Cette deuxième édition annule et remplace la première édition, parue en 2007. Cette édition constitue une révision technique.

Les principaux changements par rapport à l'édition précédente sont ceux qui suivent:

- corrections, dans le Tableau 10 et dans le Tableau 48;

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

- clarification en 6.9.1.2;
- suppression des brevets expirés et divulgation de nouveaux brevets.

La présente version bilingue (2012-07) correspond à la version anglaise monolingue publiée en 2010-08.

Le texte anglais de cette norme est issu des documents 65C/607/FDIS et 65C/621/RVD.

Le rapport de vote 65C/621/RVD donne toute information sur le vote ayant abouti à l'approbation de cette norme.

La version française de cette norme n'a pas été soumise au vote.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

Une liste de toutes les parties de la série CEI 61158, présentées sous le titre général *Réseaux de communication industriels – Spécifications de bus de terrain*, peut être consultée sur le site web de la CEI.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de la CEI sous "http://webstore.iec.ch" dans les données relatives à la publication recherchée. À cette date, la publication sera:

- reconduite;
- supprimée;
- remplacée par une édition révisée, ou
- amendée.

NOTE 2 La révision de la présente norme sera synchronisée avec les autres parties de la série CEI 61158.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# INTRODUCTION

La présente partie de la CEI 61158 est l'une d'une série produite pour faciliter l'interconnexion de composants d'un système d'automation. Elle est apparentée à d'autres normes de la série telle que définie par le modèle de référence des bus de terrain à trois couches décrit dans le rapport IEC/TR 61158-1.

Le protocole application fournit le service application en utilisant les services disponibles de la liaison de données ou autre couche immédiatement inférieure. Le but principal de la présente norme est de fournir un ensemble de règles pour la communication exprimées en termes des procédures devant être accomplies par des entités d'application (AE) d'homologues au moment de la communication. Ces règles pour la communication visent à fournir une base solide pour le développement et de servir une diversité de besoins:

- comme un guide pour les réalisateurs et les concepteurs;
- pour une utilisation dans les essais et achats d'équipements;
- comme partie intégrante d'un accord pour l'admission de systèmes dans l'environnement de systèmes ouverts;
- comme affinement pour la compréhension de communications à temps critiques au sein de l'interconnexion des systèmes ouverts (OSI)[1].

La présente norme est concernée, en particulier, par la communication et l'interfonctionnement des capteurs, des effecteurs et autres dispositifs d'automatisation. L'utilisation de la présente norme conjointement à d'autres normes positionnées dans les modèles de référence de l'OSI ou de bus de terrain permet à n'importe quelle combinaison de systèmes autrement incompatibles de fonctionner.

La Commission Électrotechnique Internationale (CEI) attire l'attention sur le fait qu'il est déclaré que la conformité avec les dispositions du présent document peut impliquer l'utilisation d'un brevet concernant les éléments de Type 3 et éventuellement d'autres types traités dans les éléments normatifs de la présente norme.

Les droits de brevets suivants pour le Type 3 ont été annoncés par [SI]:

| Publication | Titre |
|---|---|
| EP0604668-A1 (06.07.1994); EP0604668-B1 (18.02.1998) | Logical ring with monitoring of rotation time |
| EP0604669-A1 (06.07.1994); EP0604669-B1 (01.04.1998) | Bus system with monitoring of the activity state of participants |

La CEI ne prend pas de position quant à la preuve, à la validité et à la portée de ces droits de propriétés.

Le détenteur de ces droits de propriété a donné l'assurance à la CEI qu'il consent à négocier des licences avec des demandeurs du monde entier, à des termes et conditions raisonnables et non discriminatoires. A ce propos, la déclaration du détenteur des droits de propriété est enregistrée à la CEI. Des informations peuvent être demandées à:

[SI]: Siemens AG
CT IP L&T
Hr. Hans-Jörg Müller
Otto-Hahn-Ring 6

1 OSI = *Open Systems Interconnexion*

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

D-81739 Munich
Germany

L'attention est attirée sur le fait que certains des éléments du présent document peuvent faire l'objet de droits de propriété autres que ceux mentionnés ci-dessus. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et de ne pas avoir signalé leur existence.

L'ISO (www.iso.org/patents) et la CEI (http://www.iec.ch/tctools/patent_decl.htm) maintiennent à disposition des bases de données en ligne des brevets relatifs à leurs normes. Les utilisateurs sont invités à les consulter pour obtenir les dernières informations relatives à ces brevets.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

## Partie 6-3: Spécification de protocole de la couche application – Eléments de Type 3

## 1 Domaine d'application

### 1.1 Généralités

La Couche application de bus de terrain (FAL[2]) fournit aux programmes d'utilisateur un moyen d'accéder à l'environnement de communication du bus de terrain. À cet égard, la FAL peut être vue comme une «fenêtre entre des programmes d'application correspondants».

La présente norme fournit les éléments communs pour les communications de messagerie de base à temps critique et à temps non critique entre des programmes d'application dans un environnement d'automatisation et le matériau spécifique au bus de terrain de Type 3. Le terme "à temps critique" sert à représenter la présence d'une fenêtre temporelle, dans les limites de laquelle une ou plusieurs actions spécifiées sont tenues d'être parachevées avec un certain niveau défini de certitude. Le manquement à parachever les actions spécifiées dans les limites de la fenêtre temporelle risque d'entraîner la défaillance des applications qui demandent ces actions, avec le risque concomitant pour l'équipement, l'installation et éventuellement pour la vie humaine.

La présente norme définit de manière abstraite le comportement visible de l'extérieur fourni par la couche application de bus de terrain Type 3 en termes

a) de la syntaxe abstraite définissant les unités de données de protocole de couche application acheminées entre les entités d'application engagées dans une communication;

b) de la syntaxe de transfert définissant les unités de données de protocole de couche application acheminées entre les entités d'application engagées dans une communication;

c) du diagramme d'états de contexte application définissant le comportement de service application visible entre des entités d'application engagées dans une communication; et

d) des diagrammes d'états de relation d'applications définissant le comportement de communication visible entre des entités d'application engagées dans une communication;

Le but de la présente norme est de définir le protocole fourni pour

a) définir la représentation câblée des primitives de service spécifiées dans la CEI 61158-5-3, et

b) définir le comportement visible de l'extérieur qui est associé à leur transfert.

La présente norme spécifie le protocole de la couche application des réseaux de terrain de Type 3, en conformité avec le Modèle de référence de base de l'OSI (ISO/CEI 7498) et la Structure de la couche application de l'OSI (ISO/CEI 9545).

Les services et protocoles de la FAL sont fournis par des entités d'application (AE[3]) de la FAL contenues dans les processus d'application. L'AE de la FAL se compose d'un jeu d'éléments de service application (ASE[4]) orientés objet et d'une entité de gestion de couche (LME[5]) qui

---

2 FAL = *Fieldbus Application Layer*

3 AE = *Application Entity*

4 ASE = *Application Service Element*

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

gère l'AE. Les ASE fournissent des services de communication qui fonctionnent sur un jeu de classes d'objets de processus application (APO)[6]connexes. L'un des ASE de la FAL est un ASE de gestion qui fournit un jeu commun de services pour la gestion des instances de classes de la FAL.

Bien que ces services spécifient, du point de vue des applications, la manière dont la demande et les réponses sont émises et délivrées, ils n'incluent pas une spécification de ce que les applications qui demandent et qui répondent doivent en faire. À savoir, les aspects comportementaux des applications ne sont pas spécifiés; seule une définition des demandes et réponses qu'elles peuvent envoyer/recevoir est spécifiée. Cela permet une plus grande flexibilité aux utilisateurs de la FAL pour normaliser un tel comportement d'objet. En plus de ces services, certains services d'appui sont également définis dans la présente norme pour fournir l'accès à la FAL afin de maîtriser certains aspects de son fonctionnement.

### 1.2 Spécifications

L'objet principal de la présente norme est de spécifier la syntaxe et le comportement du protocole de couche application qui achemine les services de couche application définis dans la IEC 61158-5-3.

Un objectif secondaire est de fournir des trajets de migration à partir de protocoles de communications industriels préexistants. C'est ce dernier objectif qui donne naissance à la diversité des protocoles normalisés dans des parties des sous-parties de la IEC 61158-6.

### 1.3 Conformité

La présente norme ne spécifie aucune mise en œuvre ou aucun produit individuels, de même qu'elle ne restreint nullement les mises en œuvre des entités de couche application dans les systèmes d'automation industriels

Il n'est pas défini de conformité d'équipement à la présente norme de définition des services de la couche application. À la place, la conformité est obtenue par la mise en œuvre de cette spécification de protocole de couche application.

## 2 Références normatives

Les documents de référence suivants sont indispensables pour l'application du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

CEI 60559, *Arithmétique binaire en virgule flottante pour systèmes à microprocesseurs*

IEC 61158-3-3:2010[7], *Industrial communication networks – Fieldbus specifications – Part 3-3: Data-link layer service definition – Type 3 elements* (disponible uniquement en anglais)[8]

IEC 61158-4-3:2010, *Industrial communication networks – Fieldbus specifications – Part 4-3: Data-link layer protocol specification – Type 3 elements* (disponible uniquement en anglais)

IEC 61158-5-3:2010, *Industrial communication networks – Fieldbus specifications – Part 5-3: Application layer service definition – Type 3 elements* (disponible uniquement en anglais)

---

5 LME = *Layer Management Entity*

6 APO = *Application Process Object*

7 À publier.

8 Les publications monolingues des séries IEC 61158 et IEC 61784 sont actuellement en cours de traduction.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

IEC 61158-6-10:2010, *Industrial communication networks – Fieldbus specifications – Part 6-10: Application layer protocol specification – Type 10 elements* (disponible uniquement en anglais)

ISO/CEI 10731, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base – Conventions pour la définition des services OSI*

ISO/CEI 7498-1, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base: Le modèle de base*

ISO/CEI 8822, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Définition du service de présentation*

ISO/CEI 8824-1, *Technologies de l'information – Notation de syntaxe abstraite numéro un (ASN.1) : Spécification de la notation de base*

ISO/CEI 9545, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Structure de la couche application*

## 3 Termes, définitions, abréviations, symboles et conventions

### 3.1 Termes et définitions référencés

#### 3.1.1 Termes de l'ISO/CEI 7498-1

Pour les besoins du présent document, les termes suivants tels que définis dans l'ISO/CEI 7498-1 s'appliquent:

a) entité d'application
b) processus d'application
c) unité de données de protocole application
d) élément de service application
e) invocation d'entité d'application
f) invocation de processus d'application
g) transaction d'application
h) système ouvert réel
i) syntaxe de transfert

#### 3.1.2 Termes de l'ISO/CEI 8822

Pour les besoins du présent document, les termes suivants tels que définis dans l'ISO/CEI 8822 s'appliquent:

a) syntaxe abstraite
b) contexte de présentation

#### 3.1.3 Termes de l'ISO/CEI 9545

Pour les besoins du présent document, les termes suivants tels que définis dans l'ISO/CEI 9545 s'appliquent:

a) association d'applications

b) contexte d'application

c) nom de contexte d'application

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

d) invocation d'entité d'application

e) type d'entité d'application

f) invocation de processus d'application

g) type de processus d'application

h) élément de service d'application

i) élément de service de contrôle d'application

### 3.1.4 Termes de l'ISO/CEI 8824-1

Pour les besoins du présent document, les termes suivants tels que définis dans l'ISO/CEI 8824-1 s'appliquent:

a) identificateur d'objet

b) type

### 3.1.5 Termes de la couche liaison de données de réseau de terrain

Pour les besoins du présent document, les termes suivants tels que définis dans la CEI 61158-3-3 et dans la CEI 61158-4-3 s'appliquent.

a) DL-Time (heure de liaison de données)

b) DL-Scheduling-policy (politique de programmation de liaison de données)

c) DLCEP

d) DLC

e) DL-connection-oriented mode (mode orienté connexion de liaison de données)

f) DLPDU

g) DLSDU

h) DLSAP

i) link (liaison)

j) network address (adresse réseau)

k) node address (adresse de nœud)

l) node (nœud)

m) scheduled (programmé)

## 3.2 Définitions supplémentaires

**3.2.1**
**protection d'accès**
limitation de l'utilisation d'un objet d'application à un seul client

**3.2.2**
**address-assignment-table**
mise en correspondance du stockage d'objet de données E/S interne du client avec les objets de données d'entrée et de sortie décentralisées

**3.2.3**
**voie**
simple liaison physique ou logique d'un objet d'application d'entrée ou de sortie d'un serveur au processus

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**3.2.4**
**diagnostic lié à une voie**
informations concernant un élément spécifique d'un objet d'application d'entrée ou de sortie, fournies à des fins de maintenance

EXEMPLE: validité de données

**3.2.5**
**vérification de configuration**
comparaison de la structuration attendue de l'objet de données E/S côté client avec la structuration effective de l'objet de données E/S côté serveur dans la phase démarrage

**3.2.6**
**défaut de configuration**
différence inacceptable entre la structuration attendue de l'objet de données E/S et la structuration effective de l'objet de données E/S, telle que détectée par le serveur

**3.2.7**
**identificateur de configuration**
représentation d'une partie des données E/S d'un seul module d'entrée et/ou de sortie d'un serveur

**3.2.8**
**diagnostic lié à l'identificateur de configuration**
comprend toutes les données spécifiques à un module disponibles au niveau du serveur à des fins de maintenance

**3.2.9**
**commandes de commandes**
appels à action transférés d'un client à un serveur pour effacer les sorties, geler les entrées et/ou synchroniser les sorties

**3.2.10**
**cohérence de données**
moyen pour une émission et un accès cohérents de l'objet de données d'entrée ou de sortie entre client et serveur et au sein d'un client et d'un serveur

**3.2.11**
**Data-eXchange-Broadcast (DXB)**
service pour l'acheminement d'informations entre le maître DP (Classe 1) et les esclaves DP assignés déclenchant la fonctionnalité Publisher et Subscriber (éditeur et abonné) dans les esclaves DP

**3.2.12**
**Adresse DL par défaut**
valeur 126 comme valeur initiale pour l'adresse DL, qui doit être changée (par exemple: attribution d'une adresse DL par le biais du bus de terrain) avant le fonctionnement avec le maître DP (classe 1)

**3.2.13**
**diagnostic lié à un dispositif**
Comprend toutes les données liées à l'esclave DP entier disponible au niveau du serveur à des fins de maintenance

**3.2.14**
**informations de diagnostic**
toutes les données disponibles au niveau du serveur à des fins de maintenance

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**3.2.15**
**regroupement d'informations de diagnostic**
informations de diagnostic système qui sont regroupées du côté client

**3.2.16**
**maître DP (classe 1)**
dispositif de commande qui commande plusieurs esclaves DP (dispositifs de terrain)

NOTE Il s'agit habituellement d'un dispositif de commande programmable ou d'un système de commande distribué.

**3.2.17**
**maître DP (classe 2)**
dispositif de commande qui gère des données de configuration (jeux de paramètres) et des données de diagnostic d'un maître DP (Classe 1), et qui accomplit en plus toutes les capacités de communication d'un maître DP (Classe 1)

**3.2.18**
**esclave DP**
dispositif de terrain qui peut être affecté à un maître DP (Classe 1) comme fournisseur pour un échange cyclique de données E/S, en outre des fonctions et alarmes acycliques pourraient être fournies

**3.2.19**
**demande de DXB**
demande de service d'échange cyclique de données avec un service DL spécifique entre le maître DP (Classe 1) et les esclaves DP assignés déclenchant la fonctionnalité Publisher (éditeur) dans les esclaves DP

**3.2.20**
**réponse de DXB**
réponse de service d'échange cyclique de données envoyée à l'adresse de diffusion (=127) et non à l'adresse individuelle du demandeur ((maître DP (Classe 1))

**3.2.21**
**freeze**
fonction au niveau des esclaves DP pour le transfert simultané de données entre l'objet de données d'entrée et le processus

**3.2.22**
**groupe**
ensemble des esclaves DP qui accomplissent une fonction Freeze (geler) ou Sync (synchroniser)

**3.2.23**
**données E/S**
objet désigné pour être transféré de façon cyclique à des fins de traitement

**3.2.24**
**ident number**
type de dispositif maître DP (Classe 1) ou esclave DP

**3.2.25**
**indice**
adresse d'un objet au sein d'un processus d'application

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**3.2.26**
**mode isochrone**
mode de fonctionnement spécial d'un système DP qui implique tant un cycle DP constant avec un programme fixe de services DP cycliques et acycliques que la synchronisation de l'application dans le maître DP (Classe 1) et les esclaves DP avec ce cycle DP constant

**3.2.27**
**jeu de paramètres de maître**
données de configuration et de paramétrisation de tous les esclaves DP qui sont assignés au maître DP correspondant et aux paramètres de bus

**3.2.28**
**module**
unité adressable à l'intérieur d'un esclave DP

**3.2.29**
**donnée(s) de processus**
objet(s) déjà prétraité(s) et transféré(s) de façon acyclique à des fins d'informations ou de traitement ultérieur

**3.2.30**
**éditeur**
rôle d'un point d'extrémité de CR qui émet des APDU sur le réseau de terrain en vue de leur consommation par un ou plusieurs abonnés

NOTE Un éditeur peut ne pas connaître l'identité ou le nombre d'abonnés et il peut éditer ses APDU en utilisant une CR spécialisée.

**3.2.31**
**configuration réelle**
structure des données d'entrée et sortie d'un esclave DP, y compris la définition de la cohérence des données

**3.2.32**
**abonné**
rôle d'un CREP dans lequel il reçoit des unités APDU produites par un éditeur

**3.2.33**
**position**
adresse d'un module au sein d'un esclave DP

**3.2.34**
**sync**
fonction au niveau des esclaves DP pour le transfert simultané de données entre l'objet de données de sortie et le processus

### 3.3 Abréviations et symboles

| | |
|---|---|
| Ack | Acknowledgment (acquittement) |
| Add | Adresse |
| AP | Application Process (processus d'application) |
| APDU | Application Protocol Data Unit (Unité de donnée de protocole application) |
| API | Application Process Identifier (Identificateur de processus d'application) |
| AR | Application Relationship (Relation d'applications) |
| AREP | Application Relationship End Point (Point d'extrémité de relation d'applications) |
| ARL | Application Relationship List (Liste de relations d'applications) |
| ASE | Application Service Element (Élément de service application) |
| Cfg | Configuration |
| cnf | primitive "confirmation" |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| | |
|---|---|
| CREP | Communication Relationship End Point (Point d'extrémité de relation de communication) |
| CRL | Communication Relationship List (Liste de relations de communication) |
| D_ | Destination |
| DLL | Data Link Layer (Couche liaison de données) |
| DLM | Data Link-management (Gestion de liaison de données) |
| DLSAP | Data link Service Access Point (Point d'accès au service liaison de données) |
| DLSDU | Data-link Service data unit (Unité de donnée de service de liaison de données) |
| DMPM | Data Link Mapping Protocol Machine (Diagramme d'état de protocole de mapping de liaison de données) |
| DMPMM1 | Data Link Mapping Protocol Machine DP-master (Classe 1) (Maître DP (Classe 1) de Diagramme d'état de protocole de mapping de liaison de données) |
| DMPMM2 | Data Link Mapping Protocol Machine DP-master (Classe 2) (Maître DP (classe 2) de de Diagramme d'état de protocole de mapping de liaison de données) |
| DMPMS | Data Link Mapping Protocol Machine DP-slave (Esclave DP de de Diagramme d'état de protocole de mapping de liaison de données) |
| DSAP | Destination Access Point (Point d'accès de destination) |
| DXB | Data eXchange Broadcast (Diffusion d'échange de données) |
| FAL | Fieldbus Application Layer (Couche application de bus de terrain) |
| FIFO | First-In First-Out (Premier entré, premier sorti) |
| FSPMM1 | FAL Service Protocol Machine DP-master (Classe 1) (Diagramme d'état de protocole de services de la FAL Maître DP) |
| FSPMM2 | FAL Service Protocol Machine DP-master (Classe 2) (Diagramme d'état de protocol de services de la FAL Maître DP) |
| FSPMS | FAL Service Protocol Machine DP-slave (Diagramme d'état de protocol de services de la FAL Esclave DP) |
| HSA | Highest Station Address (Adresse de station la plus élevée) |
| I&M | Profil d'identification et de maintenance |
| ID | Identificateur |
| CEI | Commission Électrotechnique Internationale |
| ind | primitive "indication" |
| Inp | Input (entrée) |
| ISO | Organisation internationale de normalisation |
| IsoM | Isochronous Mode (mode isochrone) |
| L_sdu | Link Service Data Unit (Unité de donnée de service de liaison) |
| lsb | least significant bit (bit de poids faible) |
| msb | most significant bit (bit de poids fort) |
| OSI | Open Systems Interconnection (interconnexion des systèmes ouverts) |
| Outp | Output (Sortie) |
| Param | Paramètre |
| PDU | Protocol Data Unit |
| RD_ | Read (Lecture) |
| Rem | Remote (distant) |
| Req | Request (Demande, utilisée pour indiquer un type d'APDU) |
| req | primitive "request" |
| REQ | Demande pour indiquer une PDU de demande |
| RES | Réponse pour indiquer une PDU de réponse |
| RM | Resource Manager (Gestionnaire de ressources) |
| RPL_UPD | Reply Update (Mise à jour de réponse) |
| Rsp | Response (Réponse, utilisée pour indiquer un type d'APDU) |
| rsp | response primitive (primitive "response") |
| S_ | Source |
| SAP | Service Access Point (Point d'accès au service) |
| SCL | Security Level (Niveau de sécurité) |
| SD | Start Delimiter (Délimiteur de début) |
| SDN | Send Data with no Acknowledge (Envoyer des données sans acquittement) |
| SDR | Station Delay Responder (Répondeur à retard de poste) |
| Seq | Séquence |
| Src | Source |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| | |
|---|---|
| SRD | Send and Request Data with Reply (Données d'envoi et de demande avec réponse) |
| SSAP | Source Service Access Point (Point d'accès aux services de source) |
| TR | Technical Report (Rapport technique) |
| USIF | User Interface (Interface utilisateur) |
| WD | Watchdog (chien de garde) |

## 3.4 Conventions

### 3.4.1 Concept général

La couche FAL est définie comme un jeu d'éléments ASE orientés objet. Chaque ASE est spécifiée dans un paragraphe distinct. Chaque spécification d'ASE est constituée de trois parties: ses définitions de classe, ses services et sa spécification de protocole. Les deux premières parties sont contenues dans la CEI 61158-5-3. La spécification de protocole pour chacun des ASE est définie dans la présente norme.

Les définitions de classe définissent les attributs pris en charge par chaque ASE. Les attributs sont accessibles à partir d'instances de la classe en utilisant les services d'ASE de gestion ("Management") spécifiés dans la norme CEI 61158-5-3. La spécification de services définit les services qui sont fournis par l'ASE.

La présente norme utilise les conventions descriptives données dans l'ISO/CEI 10731.

### 3.4.2 Conventions de syntaxe abstraite

Les PDU sont décrites sous la forme d'octets ou de groupes d'octets.

a) Les groupes d'octets séparés par une virgule apparaissent dans l'ordre dans lequel ils sont transférés.

   Si des octets facultatifs ne sont pas présents, les octets suivants apparaissent sans trous;

b) Si des octets ou groupes d'octets sont regroupés à l'intérieur d'accolades "{ }", l'ordre est arbitraire;

c) Si des octets ou groupes d'octets sont repérés par un "*", ils peuvent apparaître plus d'une fois.
   S'ils sont utilisés à l'intérieur d'une section d'accolades "{ }", ils peuvent apparaître mélangés à d'autres octets ou à autre groupe d'octets de cette section.

d) Des octets peuvent être regroupés ou les valeurs peuvent être attribuées à l'intérieur de parenthèses "( )"

e) Si des octets ou groupes d'octets sont regroupés à l'intérieur de crochet "[ ]", le groupe peut être omis;

f) Des APDU complexes peuvent être construites au moyen de substitutions (sous-structures)

g) Les sélections exclusives d'octets ou de groupes d'octets sont séparées par un caret "^"

NOTE 1 Exemple formel de PDU

AP_PDU=Octet1,OctetGroup1,[Octet2],[Octet3],{[OctGroup2*],OctetGroup3^Octet4}
Selon ceci, les variantes suivantes sont valides sur le fil (non exhaustives):

Variante 1: Octet1, OctetGroup1, Octet2, Octet3, OctetGroup2,OctetGroup3

Variante 2: Octet1, OctetGroup1, Octet2, Octet3, OctetGroup2, OctetGroup2, OctetGroup2,OctetGroup3

Variante 3: Octet1, OctetGroup1, OctetGroup2, OctetGroup2, OctetGroup2,OctetGroup3, OctetGroup2

Variante 4: Octet1, OctetGroup1, OctetGroup2, OctetGroup3, OctetGroup2, OctetGroup2, OctetGroup2, OctetGroup2

Variante 5: Octet1, OctetGroup1, Octet3, Octet4

NOTE 2 L'ordre arbitraire implique que les groupes d'octets sont caractérisés par un en-tête spécial qui est décrit dans le cadre des règles de codage.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

NOTE 3 La syntaxe des APDU implique que conformément à la DLSDU maximale, une APDU ne doit pas dépasser 244 octets au total.

### 3.4.3 Convention pour le codage des bits et octets réservés

Le terme "reserved" (réservé) peut être utilisé pour décrire des bits dans des octets ou des octets complets. Il convient que tous les bits ou octets qui sont réservés soient mis à zéro du côté envoi et ils ne doivent pas être testés du côté réception, sauf si cela est énoncé de façon explicite ou si les bits ou octets réservés sont vérifiés par un diagramme d'états.

Le terme "reserved" (réservé) peut aussi être utilisé pour indiquer que certaines valeurs dans la plage d'un paramètre sont réservées pour des extensions futures. Dans ce cas, il convient de ne pas utiliser les valeurs réservées du côté d'envoi et elles ne doivent pas être testées du côté réception, sauf si cela est énoncé de façon explicite ou si les valeurs réservées sont vérifiées par un diagramme d'états.

### 3.4.4 Conventions pour les codages communs des octets de champs spécifiques

Les APDU peuvent contenir des champs spécifiques qui contiennent des informations d'une manière primitive et condensée. Ces champs doivent être codés dans l'ordre conforme à la Figure 1.

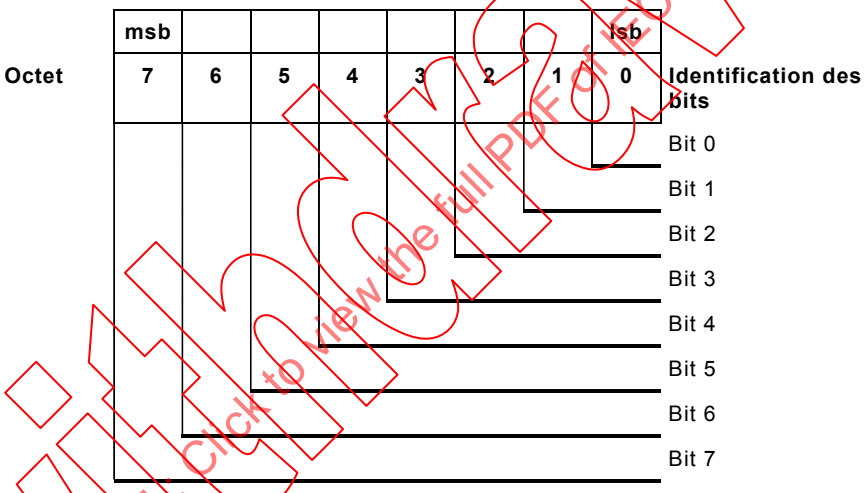



**Figure 1 – Structure commune des champs spécifiques**

- Des bits peuvent être regroupés en groupe de bits. Chaque bit ou groupe de bits doit être adressé par son Bit Identification (identification de bit) (par exemple: Bit 0, Bit 1 à 4). La position dans un octet doit être conforme à la figure ci-dessus. Des pseudonymes peuvent être utilisés pour chaque bit ou groupe de bits ou bien ils peuvent être marqués comme étant réservés. Le regroupement de bits individuels doit être dans l'ordre croissant sans trous. Les valeurs d'un groupe de bits peuvent être représentées comme des valeurs binaires, décimales ou hexadécimales. Cette valeur doit être seulement valide pour les bits regroupés et peut seulement représenter l'octet tout entier si tous les 8 bits sont regroupés. Les valeurs décimales ou hexadécimales doivent être transformées en valeurs binaires afin que le bit ayant le numéro de groupe le plus élevé représente le msb concernant les bits groupés.

EXEMPLE 1: Description et relation pour l'octet de champ spécifique

Bit 0: reserved.

Bit 1à 3: Reason_Code La valeur décimale 2 pour le Reason_Code signifie "erreur générale"

Bit 4 à 7: doit toujours être mis à 1.

L'octet qui est construit conformément à la description ci-dessus ressemble à ce qui suit:

(msb) Bit 7 = 1,

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

Bit 6 = 1,

Bit 5 = 1,

Bit 4 = 1,

Bit 3 = 0,

Bit 2 = 1,

Bit 1 = 0,

(lsb) Bit 0 = 0.

La combinaison de bits "0-1-0" pour les bits 1 à 3 est égale à la valeur décimale 2.

### 3.5 Conventions utilisées dans les diagrammes d'états

#### 3.5.1 Conventions dans les diagrammes d'états

Les séquences de protocoles sont décrites au moyen de diagrammes d'états.

Dans les diagrammes d'états, les états sont représentés sous forme de boîtes, tandis que les transitions d'états sont montrées comme des flèches. Les noms des états et des transitions du diagramme d'états correspondent aux noms dans l'énumération textuelle des transitions d'états.

L'énumération textuelle des transitions d'états est structurée comme suit (voir aussi le Tableau 1).

La première colonne contient le nom de la transition.

La deuxième colonne définit l'état courant.

La troisième colonne contient un événement facultatif suivi de Conditions commençant par une barre oblique "/" comme premier caractère de ligne et, pour finir, suivi par les Actions commençant par un "=>" comme premier caractère de ligne.

La dernière colonne contient le prochain état.

Si l'événement se produit et les conditions sont remplies, la transition se lance, c'est-à-dire que les actions sont exécutées et il y a entrée dans le prochain état.

La présentation d'un Diagramme est montrée dans le Tableau 1. La signification des éléments d'une Description de diagramme d'états est montrée dans le Tableau 2.

**Tableau 1 – Éléments de description de diagramme d'états**

| # | État courant | Événement ou condition => action | Prochain état |
|---|---|---|---|
| | | | |

**Tableau 2 – Description d'éléments de diagramme d'états**

| Élément de description | Signification |
|---|---|
| État courant<br>Prochain état | Nom des états donnés |
| # | Nom ou numéro de la transition d'état. |
| Événement | Nom ou description de l'événement. |
| /Condition | Expression booléenne. Le symbole "/" qui précède ne fait pas partie de la condition. |
| => Action | Liste des assignations et invocations de service ou de fonction. Le symbole "=>" qui précède ne fait pas partie de l'action. |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Les conventions utilisées dans les diagrammes d'états sont montrées dans le Tableau 3.

**Tableau 3 – Conventions utilisées dans les diagrammes d'états**

| Convention | Signification |
|---|---|
| := | La valeur d'un élément à gauche est remplacée par la valeur d'un élément à droite. Si un élément à droite est un paramètre, il provient de la primitive montrée comme un événement d'entrée. |
| xxx | Un nom de paramètre.<br>Exemple:<br>Identifier:= reason<br>signifie que la valeur d'un paramètre 'reason' est attribuée à un paramètre appelé 'Identifier.' |
| "xxx" | Indique une valeur fixe.<br>Exemple:<br>Identifier:= "abc"<br>signifie que la valeur "abc" est attribuée à un paramètre nommé 'Identifier.' |
| = | Une condition logique pour indiquer qu'un élément à gauche est égal à un élément à droite. |
| < | Une condition logique pour indiquer qu'un élément à gauche est inférieur à un élément à droite. |
| > | Une condition logique pour indiquer qu'un élément à gauche est supérieur à un élément à droite. |
| <> | Une condition logique pour indiquer qu'un élément à gauche n'est pas égal à un élément à droite. |
| && | "AND" (c'est-à-dire: ET) logique |
| \|\| | "OR" (c'est-à-dire: OU) logique |
| for (Identifier:= start_value to end_value)<br>actions<br>endfor | Cette construction permet l'exécution d'une séquence d'actions dans une boucle au sein d'une même transition. La boucle est exécutée pour toutes les valeurs de start_value à end_value. |
| If (condition)<br>actions<br>else<br>actions | Cette construction permet l'exécution d'actions alternatives dépendant d'une certaine condition (qui pourrait être la valeur d'un certain identificateur ou le résultat d'une action précédente) au sein d'une même transition.<br>Les parties commençant par “else” peuvent être omises s'il n'y a aucune action si la condition n'est pas remplie. |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Il est fortement recommandé aux lecteurs de se référer aux paragraphes concernant les définitions des attributs AREP et CREP, aux fonctions locales et aux définitions des FAL-PDU pour appréhender les diagrammes d’état de protocole. Les lecteurs sont censés avoir une connaissance suffisante de ces définitions et celles-ci sont utilisées sans autres explications.

En outre, les éléments de description suivants sont utilisés:

**Caractères génériques dans les noms**

name_XXX: “XXX” est utilisé comme chaîne générique pour tous les noms commençant par “name”.

L'utilisation typique d'un caractère générique l’est pour un Event (événement). Dans ce contexte, il y a autant de transitions d'états qu'il existe d'événements possibles pour ce caractère générique.

**Macro conditionnelle**

<CONDITIONAL -MACRO-NAME>

<

CONDITION1: macrobody1

CONDITION2: macrobody2

…

>

CONDITION1, CONDITION2, etc. définissent tous les cas possibles pour la macro conditionnelle. L'expression “CONDITIONAL-MACRO-NAME” sert de réceptacle pour le “macrocode” dépendant du résultat de la condition.

**Macro de remplacement**

<REPLACEMENT-MACRO-NAME>

<

XXX= Name1: macrobody1

XXX= Name2: macrobody2

…

>

Name1, Name2, etc. définissent tous les cas possibles pour la macro de remplacement. L'expression “REPLACEMENT-MACRO-NAME” sert de réceptacle pour le “macrocode” dépendant de la valeur de l'utilisation courante du caractère générique d XXX.

Exemple:
<SERVICE_REQ_PARA>
<
XXX=Read: Para1, Para2
XXX=Write: Para1, Para2, Para3
>
Lorsqu'elle est appelée dans
MSAB_XXX.req(<SERVICE_REQ_PARA>)
elle donnera
MSAB_Read.req(Para1, Para2) ou MSAB_Write.req(Para1, Para2, Para3)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# 4 Description de la syntaxe de FAL

## 4.1 Syntaxe abstraite des APDU

Le Tableau 4 définit la syntaxe abstraite des PDU de couche application de DP appelées "des APDU". L'ordre défini des octets doit être utilisé pour acheminer les APDU.

**Tableau 4 – Syntaxe des APDU**

| Nom de l'APDU | Structure d'une APDU |
|---|---|
| DataExchange-REQ-PDU | [Outp_Data*] |
| DataExchange-RES-PDU | [Inp_Data*] |
| RD_Input-RES-PDU | [Inp_Data*] |
| RD_Output-RES-PDU | [Outp_Data*] |
| Chk_Cfg-REQ-PDU | {[Identifier_Format*], [Special_Identifier_Format*]^ [Extended_Special_Identifier_Format*]} |
| Get_Cfg-RES-PDU | {[Identifier_Format*], [Special_Identifier_Format*]^[Extended_Special_Identifier_Format*]} |
| Set_Prm-REQ-PDU | Station_status, WD_Fact_1, WD_Fact_2, min_TSDR, Ident_Number, Group_Ident, User_Prm_Data |
| Set_Ext_Prm-REQ-PDU | Structured_Prm_Data* |
| Diagnosis-RES-PDU | Station_status_1, Station_status_2, Station_status_3, Diag_Master_Add, Ident_Number, {[Device_Related_Diagnosis*]^([Alarm], [Status*],[Modul_Status],[DXB_Link_Status], { [Red_Status] ^[PrmCmdAck]}), [Identifier_Related_Diagnosis], [Channel_Related_Diagnosis*], [Revision_Number]}<br>Le champ Device_Related_Diagnosis peut être présent si DPV1=FALSE, autrement si DPV1=TRUE, il ne doit pas être présent. Dans ce cas, les champs Alarm, Status, Modul_Status peuvent être présents. |
| Set_Slave_Add-REQ-PDU | New_Slave_Add, Ident_Number, No_Add_Chg, [Rem_Slave_Data*] |
| Global_Control-REQ-PDU | Control_Command, Group_Select |
| Clock-Value-PDU | Clock_value_time_event, Clock_value_previous_TE, Clock_value_status1, Clock_value_status2 |
| Initiate-REQ-PDU | Function_Num(0x57), reserved (3 Octets), Send_Timeout, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Para |
| Initiate-RES-PDU | Function_Num(0x57), Max_Len_Data_Unit, Features_Supported, Profile_Features_Supported, Profile_Ident_Number, Add_Addr_Param |
| Initiate-NRS-PDU | Function_Num(0xD7), Error_Decode, Error_Code_1, Error_Code_2 |
| Abort-REQ-PDU | Function_Num(0x58), Subnet, Instance_Reason_Code |
| Read-REQ-PDU | Function_Num(0x5E), Slot_Number, Index (0 .. 254), Length |
| Read-RES-PDU | Function_Num(0x5E), Slot_Number, Index (0 .. 254), Length, [Data*] |
| Read-NRS-PDU<br>Pull-NRS-PDU | Function_Num(0xDE), Error_Decode, Error_Code_1, Error_Code_2 |
| Write-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (0 .. 254), Length, [Data*] |
| Write-RES-PDU | Function_Num(0x5F), Slot_Number, Index (0 .. 254), Length |
| Write-NRS-PDU<br>Initiate_Load_NRS-PDU<br>Push-NRS-PDU<br>Terminate_Load-NRS-PDU<br>Start-NRS-PDU<br>Stop-NRS-PDU<br>Resume-NRS-PDU<br>Reset-NRS-PDU<br>Call-NRS-PDU<br>Get_FI_State-NRS-PDU | Function_Num(0xDF), Error_Decode, Error_Code_1, Error_Code_2 |
| Alarm_Ack-REQ-PDU | Function_Num(0x5C), Slot_Number, Alarm_Type, Alarm_Specifier |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de l'APDU | Structure d'une APDU |
|---|---|
| Alarm_Ack-RES-PDU | Function_Num(0x5C), Slot_Number, Alarm_Type, Alarm_Specifier |
| Alarm_Ack-NRS-PDU | Function_Num(0xDC), Error_Decode, Error_Code_1, Error_Code_2 |
| Idle-REQ-PDU | Function_Num(0x48) |
| Idle-RES-PDU | Function_Num(0x48) |
| Data_Transport-REQ-PDU | Function_Num(0x51), Slot_Number, Index, Length, [Data*] |
| Data_Transport-RES-PDU | Function_Num(0x51), Slot_Number, Index, Length, [Data*] |
| Data_Transport-NRS-PDU | Function_Num(0xD1), Error_Decode, Error_Code_1, Error_Code_2 |
| Initiate_Load-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x00), Options, LR_Index, LR_Length, Intersegment_Request_Timeout, [User_Specific] |
| Initiate_Load-RES-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x00), Max_Segment_Length, LR_Index, LR_Length, Max_Response_Delay |
| Push-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x01), Options, Sequence_Number, LR_Data |
| Pull-REQ-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length |
| Pull-RES-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x02), Options, Sequence_Number, LR_Data |
| Terminate_Load-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x03), reserved (1 octet), reserved (2 Octets) |
| Start-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x04), reserved (1 octet), FI_Index, [Execution_Argument] |
| Stop-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x05), reserved (1 octet), FI_Index |
| Resume-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x06), reserved (1 octet), FI_Index, [Execution_Argument] |
| Reset-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x07), reserved (1 octet), FI_Index |
| Call-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x08), Entity Number, FI_Index (=0…64999), [Execution_Argument]<br>Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x08), Entity Number, FI_Index (=65000..65199), [I MData_Exexcution_Argument] |
| Call-RES-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x08), Entity Number, FI_Index (=0…64999), [Result_Argument]<br>Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x08), Entity Number, FI_Index (=65000…65199), [IMData_Result_Argument] |
| Get_FI_State-REQ-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length, Extended_Function_Num (0x09), reserved (1 octet), FI_Index |
| Get_FI_State-RES-PDU | Function_Num(0x5E), Slot_Number, Index (255), Length, Extended_Function_Num (0x09), reserved (1 octet), FI_Index, FI_State |
| Push-RES-PDU<br>Terminate_Load-RES-PDU<br>Start-RES-PDU<br>Stop-RES-PDU<br>Resume-RES-PDU<br>Reset-RES-PDU | Function_Num(0x5F), Slot_Number, Index (255), Length |
| RM-REQ-PDU | Function_Num(0x56), Server_SAP, Send_Timeout |
| Get_Master_Diag-REQ-PDU | Function_Num(0x41), MDiag_Identifier |
| Get_Master_Diag-RES-PDU | Function_Num(0x41), MDiag_Identifier(=0..125), Diagnosis-RES-PDU<br>Function_Num(0x41), MDiag_Identifier(=126), System_Diagnosis,<br>Function_Num(0x41), MDiag_Identifier(=127), Master_Status<br>Function_Num(0x41), MDiag_Identifier(=128), Data_Transfer_List |
| Start_Seq-REQ-PDU | Function_Num(0x42), Area_CodeUpDownload, Timeout |
| Start_Seq-RES-PDU | Function_Num(0x42), Area_CodeUpDownload, Timeout, Max_Len_Data_Unit |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de l'APDU | Structure d'une APDU |
|---|---|
| Download-REQ-PDU | Function_Num(0x43), Area_CodeUpDownload(=0..125), Add_Offset, DP_Slave_Parameter_Set<br>Function_Num(0x43), Area_CodeUpDownload(=127), Add_Offset, Bus_Parameter_Set<br>Function_Num(0x43), Area_CodeUpDownload(=129), Add_Offset, Statistic_Counters<br>Function_Num(0x43), Area_CodeUpDownload(=136 to 139), Add_Offset, Master_Parameter_Set |
| Download-RES-PDU | Function_Num(0x43), Area_CodeUpDownload, Add_Offset |
| Upload-REQ-PDU | Function_Num(0x44), Area_CodeUpDownload, Add_Offset, Data_Len |
| Upload-RES-PDU | Function_Num(0x44), Area_CodeUpDownload(=0..125), Add_Offset, DP_Slave_Parameter_Set<br>Function_Num(0x44), Area_CodeUpDownload(=127), Add_Offset, Bus_Parameter_Set<br>Function_Num(0x44), Area_CodeUpDownload(=129), Add_Offset, Statistic_Counters<br>Function_Num(0x43), Area_CodeUpDownload(=136 to 139), Add_Offset, Master_Parameter_Set |
| End_Seq-REQ-PDU | Function_Num(0x45) |
| End_Seq-RES-PDU | Function_Num(0x45) |
| Act_Para_Brct-REQ-PDU | Function_Num(0x46), Area_CodeActBrct |
| Act_Param-REQ-PDU | Function_Num(0x47), Area_CodeAct, Activate |
| Act_Param-RES-PDU | Function_Num(0x47), Area_CodeAct, Activate |
| ZERO-PDU | Octet1(0) |
| NULL-PDU | |

NOTE 1 Si une APDU est constituée seulement d'octets de données utilisateur, la distinction est faite en utilisant des points d'accès au service de couche liaison de données différents (voir Diagrammes protocolaires).

NOTE 2 Une ZERO-PDU (c'est-à-dire: PDU zéro) contient un seul octet avec tous les bits mis à zéro. Cette APDU est utilisée pour indiquer le diagnostic à partir d'un dispositif sans entrées.

NOTE 3 Une NULL-PDU (c'est-à-dire: PDU vide) ne contient pas d'octets. Elle est utilisée pour ne présenter aucune DLSDU à la couche liaison de données.

Le Tableau 5 définit des structures pour les substitutions des éléments des structures d'APDU montrées dans le Tableau 4.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 5 – Substitutions**

| Nom de substitution | Structure |
|---|---|
| Identifier_Format | Cfg_Identifier |
| Special_Identifier_Format | Special_Cfg_Identifier, [Length_Octet] , [Length_Octet], [Manufacturer_Specific_Data*]<br><br>Les champs Length_Octet doivent toujours commencer par les champs qui indiquent des données de sortie, si elles sont présentes. |
| Extended_Special_Identifier_ Format | Special_Cfg_Identifier,[ Extended_Length_Octet], [Extended_Length_Octet], [Data_Type*], [Manufacturer_Specific_Data*]<br><br>Les champs Extended_Length_Octet doivent toujours commencer par les champs qui indiquent des données de sortie, si elles sont présentes.<br><br>Le champ Data_Type ne doit être omis que dans le cas d'une position vide. Autrement, il doit être présent en rapport avec les champs Length. |
| Device_Related_Diagnosis | Header_Octet, Diagnosis_User_Data* |
| Identifier_Related_Diagnosis | Header_Octet, Identifier_Diagnosis_Data_Array |
| Channel_Related_Diagnosis | Identifier_Number, (Channel_Number, Type_of_Diagnosis)* |
| Alarm | Header_Octet, Alarm_Type, Slot_Number, Alarm_Specifier, [Diagnosis_User_Data*] |
| Status | Header_Octet, Status_Type, Slot_Number, Status_Specifier, Diagnosis_User_Data*] |
| Modul_Status | Header_Octet, Status_Type(=Modul_Status), Slot_Number(=0), Status_Specifier, Modul_Status_Array |
| DXB_Link_Status | Header_Octet, Status_Type(=DXB_Link_Status), Slot_Number(=0), Status_Specifier, (Publisher_Address, Publisher_Status)* |
| PrmCmdAck | Header_Octet, Status_Type(=PrmCmdAck), Slot_Number(=0), Status_Specifier, Function, Red_Status1, Red_Status2, Red_Status3 |
| Red_Status | Header_Octet, Status_Type(=Red_Status), Slot_Number(=0), Status_Specifier, Function, Red_Status1, Red_Status2, Red_Status3 |
| User_Prm_Data | [DPV1_Status_1, DPV1_Status_2, DPV1_Status_3], [User_Prm_Data_Element*]^[Structured_Prm_Data*]<br><br>Les champs DPV1_Status_1, DPV1_Status_2 et DPV1_Status_3 doivent être présents si DPV1=TRUE.<br><br>Structured_Prm_Data doit être présent si DPV1_Status_3.Prm_Structure est TRUE, autrement il ne doit pas être utilisé. |
| Structured_Prm_Data | Structure_Length, Structure_Type, Slot_Number, reserved (1 octet), User_Prm_Data*<br><br>Special Case DXB LinkTable:<br><br>Structure_Length, Structure_Type(=3), Slot_Number(=0), reserved (1 octet), Version(=1), (Publisher_Addr, Publisher_Length, Sample_Offset, Sample_Length)*<br><br>Special Case DXB-Subscribertable:<br><br>Structure_Length, Structure_Type(=7), Slot_Number(=0), reserved (1 octet), Version(=1), (Publisher_Addr, Publisher_Length, Sample_Offset, Dest_Slot_Number, Offset_Data_Area, Sample_Length)*<br><br>Special Case IsoM Parameter<br><br>Structure_Length, Structure_Type(=4), Slot_Number(=0), reserved (1 octet), Version(=1), TBASE_DP, TDP, TMAPC, TBASE_IO, TI, TO, TDX, TPLL_W, TPLL_D<br><br>Special Case PrmCmd<br><br>Structure_Length, Structure_Type(=2), Slot_Number(=0), Specifier, Function, Properties, Output_Hold_Time<br><br>Special Case Time AR Parameter:<br><br>Structure_Length, Structure_Type(=8), Slot_Number(=0), reserved (1 octet), Clock Sync Interval, [CS Delay Time]<br><br>Special Case F-Parameter<br><br>Structure_Length, Structure_Type(=5), Slot_Number, reserved (1 octet), User_Prm_Data* |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| Nom de substitution | Structure |
|---|---|
| Features_Supported | Features_Supported_1, Features_Supported_2 |
| Profile_Features_Supported | Profile_Features_Supported_1, Profile_Features_Supported_1 |
| Add_Addr_Para | S_Type, S_Len, D_Type, D_Len, S_API, S_SCL, [S_Network_Address], [S_MAC_Address], D_API, D_SCL, [D_Network_Address], [D_MAC_Address] |
| Master_Status | USIF_State, Ident_Number, Hardware_Release_DP, Firmware_Release_DP, Hardware_Release_User, Firmware_Release_User, reserved (9 Octets) |
| DP_Slave_Parameter_Set | Slave_Para_Len, SI_Flag, Slave_Type, Max_Diag_Data_Len, Max_Alarm_Len, Max_Channel_Data_Length, Diag_Upd_Delay, Alarm_Mode, Add_SI_Flag, MS1_Timeout, reserved (4 Octets), Prm_Data_Len, Prm_Data, Cfg_Data_Len, Cfg_Data, Add_Tab_Len, Add_Tab, Slave_User_Data_Len, Slave_User_Data, Ext_Prm_Data_Len, [Ext_Prm_Data] |
| Bus_Parameter_Set | Bus_Para_Len, DL_Add, Data_rate, $T_{SL}$, min $T_{SDR}$, max $T_{SDR}$, $T_{QUI}$, $T_{SET}$, $T_{TR}$, G, HSA, max_retry_limit, Bp_Flag, Min_Slave_Interval, Poll_Timeout, Data_Control_Time, Alarm_Max, Max_User_Global_Control, reserved (4 Octets), Master_User_Data_Len, Master_Class2_Name, Master_User_Data*, $T_{CT}$, max$T_{SH}$ |
| Master_Parameter_Set | Bus_Parameter_Set, DP_Slave_Parameter_Set* |
| Statistic_Counters | Counter_Group* |
| Counter_Group | si Add_Offset 0..126: DLPDU_sent_count(Add_Offset), Error_count(Add_Offset)<br>si Add_Offset 127: SD_sent_count, SD_error_count |
| Add_Tab | [Number_of_Entries], [Add_Tab_Entry_Header, I/O_Data_Length, IO_Config_Address, Host_Address]* |
| IMData_Execution_Argument | [I&M1] ^ [I&M2] ^ [I&M3] ^ [I&M4] ^ [I&M_Profile_Specific_x] ^ [I&M_Manufacturer_Specific_x] |
| IMData_Result_Argument | I&M0 ^ [I&M1] ^ [I&M2] ^ [I&M3] ^ [I&M4] ^ [I&M_Profile_Specific_x] ^ [I&M_Manufacturer_Specific_x] |
| I&M0 | IM_Header, IM_Manufacturer_ID, OrderID, IM_Serial_Number, IM_Hardware_Revision, IM_Software_Revision, IM_Revision_Counter, IM_Profile_ID, IM_Profile_Specific_Type, IM_Version, IM_Supported |
| I&M1 | IM_Header, IM_Tag_Function, IM_Tag_Location |
| I&M2 | IM_Header, IM_Date |
| I&M3 | IM_Header, IM_Descriptor |
| I&M4 | IM_Header, IM_Signature |
| I&M_Profile_Specific_x | IM_Header, IM_Profile_Specific_Data |
| IM_Version | IM_Version_Major, IM_Version_Minor |
| IM_Software_Revision | SWRevisionPrefix,IM_SWRevision_Functional_Enhancement, IM_SWRevision_Bug_Fix, IM_SWRevision_Internal_Change |

NOTE Le DP_Slave_Parameter_Set et le Bus_Parameter_Set peuvent dépasser la taille maximale d'APDU jusqu'à une taille de 64 Kilooctets. L'acheminement des données est donc segmenté au moyen d'une fenêtre adressée par le paramètre Add_Offset.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

## 4.2 Types de données

### 4.2.1 Notation pour le type Boolean (booléen)

```
Boolean::= BOOLEAN                    -- TRUE si la valeur n'est pas zéro.
                                      -- FALSE si la valeur est zéro.
```

#### 4.2.2 Notation pour le type Integer (entier)

Integer8::= INTEGER (-128..+127) -- plage $-2^7 <= I <= 2^7-1$

Integer16::= INTEGER (-32768..+32767) -- plage $-2^{15} <= I <= 2^{15}-1$

Integer32::= INTEGER -- plage $-2^{31} <= I <= 2^{31}-1$

Integer64::= INTEGER -- plage $-2^{63} <= I <= 2^{63}-1$

#### 4.2.3 Notation pour le type Unsigned (non signé)

Unsigned8::= INTEGER (0..255) -- plage $0 <= I <= 2^8-1$

Unsigned16::= INTEGER (0..65535) -- plage $0 <= I <= 2^{16}-1$

Unsigned32::= INTEGER -- plage $0 <= I <= 2^{32}-1$

Unsigned64::= INTEGER -- plage $0 <= I <= 2^{64}-1$

#### 4.2.4 Notation pour le type Floating Point (virgule flottante)

Floating32::= BIT STRING SIZE (4) -- Single precision (en simple précision) de la CEI 60559

Floating64::= BIT STRING SIZE (8) -- Double precision (en double précision) de la CEI 60559

#### 4.2.5 Notation pour le type OctetString (chaîne d'octets)

OctetString::= OCTET STRING -- Pour utilisation générique

#### 4.2.6 Notation pour le type VisibleString (chaîne visible)

VisibleString2::= VISIBLE STRING -- Pour utilisation générique

#### 4.2.7 Notation pour le type BinaryDate (date binaire)

BinaryDate::= OctetString7

#### 4.2.8 Notation pour le type TimeOfDay (heure du jour)

TimeOfDay avec indication de date::= OctetString6

TimeOfDay sans indication de date::= OctetString4

#### 4.2.9 Notation pour le type TimeDifference (différence horaire)

TimeDifference avec indication de date::= OctetString6

TimeDifference sans indication de date::= OctetString4

#### 4.2.10 Notation pour le type Network Time (temps réseau)

Network Time::= OctetString8

#### 4.2.11 Notation pour le type Network Time Difference (différence de temps réseau)

Network Time Difference::= OctetString8

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# 5 Syntaxe de transfert

## 5.1 Codage des types de données de base

### 5.1.1 Codage d'une valeur Boolean

a) Le codage d'une valeur Boolean doit être primitif. Les ContentsOctets doivent consister en un seul octet.

b) Si la valeur Boolean est FALSE, les ContentsOctets doivent être 0 (zéro). Si la valeur Boolean RUE, les ContentsOctets doivent être 0xff.

### 5.1.2 Codage d'une valeur Integer

a) Le codage d'une valeur Integer de longueur fixe des types Integer8, Integer16 et Integer32 doit être primitif et les ContentsOctets doivent respectivement consister exactement en un octet, deux octets et quatre octets.

b) Les ContentsOctets doivent être un nombre binaire en complément à deux égal à la valeur entière et consister en les bits 7 à 0 du premier octet, suivis des bits 7 à 0 du deuxième octet, suivis des bits 7 à 0 de chaque octet à tour de rôle jusqu'au dernier octet inclus des ContentsOctets.

NOTE La valeur du nombre binaire en complément à deux est dérivée de la numérotation des bits dans les ContentsOctets, en commençant par le bit 0 du dernier octet comme bit zéro et en terminant la numérotation par le bit 7 du premier octet. Chaque bit se voit attribuer une valeur numérique de $2^N$, où $N$ est sa position dans la séquence de numérotation ci-dessus. La valeur du nombre binaire en complément à deux est obtenue en ajoutant les valeurs numériques attribuées à chaque bit pour les bits qui sont mis à un, à l'exclusion du bit 7 du premier octet, puis en réduisant la valeur par la valeur numérique attribuée au bit 7 du premier octet si ce bit est mis à 1.

### 5.1.3 Codage d'une valeur Unsigned

a) Le codage d'une valeur Unsigned de longueur fixe des types Unsigned8, Unsigned16 et Unsigned32 doit être primitif et les ContentsOctets doivent respectivement consister exactement en un octet, deux octets et quatre octets.

b) Les ContentsOctets doivent être un nombre binaire égal à la valeur Unsigned et consister en les bits 7 à 0 du premier octet, suivis des bits 7 à 0 du deuxième octet, suivis des bits 7 à 0 de chaque octet à tour de rôle jusqu'au dernier octet inclus des ContentsOctets.

NOTE La valeur du nombre binaire est dérivée de la numérotation des bits dans les ContentsOctets, en commençant par le bit 0 du dernier octet comme bit zéro et en terminant la numérotation par le bit 7 du premier octet. Chaque bit se voit attribuer une valeur numérique de $2^N$, où $N$ est sa position dans la séquence de numérotation ci-dessus La valeur du nombre binaire est obtenue en ajoutant les valeurs numériques attribuées à chaque bit pour les bits qui sont mis à un.

### 5.1.4 Codage d'une valeur Floating-Point

a) Le codage d'une valeur Floating32 doit être primitif et les ContentsOctets doivent consister exactement en quatre octets.

b) Les ContentsOctets doivent contenir des valeurs en virgule flottante définies conformément à la CEI 60559. Le signe est codé dans le bit 7 du premier octet. Il est suivi de l'exposant commençant à partir du bit 6 du premier octet et ensuite de la mantisse commençant à partir du bit 6 du deuxième octet pour le type Floating32.

### 5.1.5 Codage d'une valeur Visible String

a) Le codage d'une valeur VisibleString de longueur variable doit être primitif.

b) Il n'y a pas de champ Length; la longueur est codée implicitement.

c) Les ContentsOctets doivent être une séquence d'octets. L'élément de chaîne le plus à gauche est codé dans le premier octet, suivi du deuxième octet, suivi de chaque octet à tour de rôle jusqu'au dernier octet inclus comme le plus à droite des ContentsOctets.

### 5.1.6 Codage d'une valeur Octet String

a) Le codage d'une valeur OctetString de longueur variable doit être primitif.

b) Il n'y a pas de champ Length; la longueur est codée implicitement.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

c) Les ContentsOctets doivent être une séquence d'octets. L'élément de chaîne le plus à gauche est codé dans le premier octet, suivi du deuxième octet, suivi de chaque octet à tour de rôle jusqu'au dernier octet inclus comme le plus à droite des ContentsOctets.

### 5.1.7 Codage d'une valeur BinaryDate

a) Le codage d'une valeur BinaryDate doit être primitif.

b) Les ContentsOctets doivent avoir une valeur égale à celle des octets dans la valeur de données, comme montré à la Figure 2:

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | 0 ms ...59 999 ms |
| 2 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| 3 | RSV | RSV | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 0 ...59 minutes |
| 4 | SU | RSV | RSV | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 0...23 heures |
| | jour de semaine | | | jour de mois | | | | | 1...7 jour de semaine |
| 5 | $2^{2}$ | $2^{1}$ | $2^{0}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 1...31 jour de mois |
| 6 | RSV | RSV | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 1...12 mois |
| 7 | RSV | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | 0 ... 50 années 2000 à 2050<br>51 ... 99 années 1951 à 1999 |
| | msb | | | | | | | | |

NOTE Pour prévenir le problème de l'An 2000, les valeurs de l'année vont de 0 à 50 pour les années 2000 à 2050 et de 51 à 99 pour les années 1951 à 1999. Cela est conforme à la Standards Institution (BSI) DISC PD2000-1:1998 “A Definition of Year 2000 Conformance Requirements” Rule 3 (b).

**Figure 2 – Codage du type de données BinaryDate**

### 5.1.8 Codage d'une valeur TimeOfDay avec et sans indication de date indication

a) Le codage d'une valeur TimeOfDay avec et sans indication de date indication doit être primitif.

b) Les ContentsOctets doivent avoir une valeur égale à celle des octets dans la valeur de données, comme montré à la Figure 3:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | 0 | 0 | 0 | 0 | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | nombre de |
| 2 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | millisecon-des |
| 3 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | à partir de |
| 4 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | minuit |
| 5 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | nombre de jours depuis le 01.01.84 |
| 6 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | uniquement avec indication de date |
| | msb | | | | | | | | |

**Figure 3 – Codage d'une valeur TimeOfDay**

### 5.1.9 Codage d'une valeur Time Difference avec et sans indication de date

a) Le codage d'une valeur Time Difference avec et sans indication de date doit être primitif.

b) Les ContentsOctets doivent avoir une valeur égale à celle des octets dans la valeur de données, comme montré à la Figure 4:

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | |
| 2 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | millisecondes |
| 3 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | |
| 4 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| 5 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | jours |
| 6 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | uniquement avec indication de date |
| | msb | | | | | | | | |

**Figure 4 – Codage d'une valeur Time Difference**

### 5.1.10 Codage d'une valeur Network Time

a) Le codage d'une valeur Network Time doit être primitif.

b) Les ContentsOctets doivent avoir une valeur égale à celle des octets dans la valeur de données, comme montré à la Figure 5:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | Secondes à partir du 1.1.1900 0.00,00 ou depuis le 7.2.2036 6.28,16 lorsque la valeur de temps est inférieure à 0x9dff4400.00000000 |
| 2 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | |
| 3 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | |
| 4 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| 5 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | Partie fractionnaire de secondes une unité est 1/(2^32) s |
| 6 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | |
| 7 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | |
| 8 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| | msb | | | | | | | | |

**Figure 5 – Codage d'une valeur Network Time**

### 5.1.11 Codage d'une valeur Network Time Difference

a) Le codage d'une valeur Network Time Difference doit être primitif.

b) Les ContentsOctets doivent avoir une valeur égale à celle des octets dans la valeur de données, comme montré à la Figure 6:

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | Secondes comme Integer32 |
| 2 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | |
| 3 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | |
| 4 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| 5 | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | Partie fractionnaire de secondes une unité est 1/(2^32) s |
| 6 | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | |
| 7 | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | |
| 8 | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | |
| | msb | | | | | | | | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Figure 6 – Codage d'une valeur Network Time Difference**

### 5.1.12 Codage d'une valeur Null

a) Le codage d'une valeur NULL doit être primitif.

b) Le ContentsOctet doit être omis.

## 5.2 Section de codage relative aux PDU d'échange de données

### 5.2.1 Codage du champ Outp_Data

L'un des types de données suivants doit être utilisé pour chaque champ Outp_Data:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.2.2 Codage du champ Inp_Data

L'un des types de données suivants doit être utilisé pour chaque champ Inp_Data:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

NOTE Les types de données avec longueur variable (Visible String, Octet String, TimeOfDay, Time Difference) pour Inp_Data ou Outp_Data ne sont présents qu'une seule fois dans DataExchange-RES-PDU ou DataExchange-REQ-PDU. Les données utilisateur correspondent au contenu de Configuration-PDU.

## 5.3 Section de codage relative aux PDU d'échange de diagnostic d'esclave

### 5.3.1 Codage du champ Station_status_1

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: Diag.Station_Non_Existent**
Doit être mis à zéro dans le cas de Diagnoses-RES-PDU

Valeur (0): signifie FALSE si Get_Master_Diag-RES-PDU

Valeur (1): signifie TRUE si Get_Master_Diag-RES-PDU

**Bit 1: Diag.Station_Not_Ready**
FALSE (0), TRUE (1)

**Bit 2: Diag.Cfg_Fault (Configuration Fault, défaut de configuration)**
FALSE (0), TRUE (1)

**Bit 3: Diag.Ext_Diag (Extended Diagnosis, diagnostic étendu)**
FALSE (0), TRUE (1)

**Bit 4: Diag.Not_Supported**
FALSE (0), TRUE (1)

**Bit 5: Diag.Invalid_Slave_Response**
Doit être mis à zéro si Diagnoses-RES-PDU

Valeur (0): signifie FALSE si Get_Master_Diag-RES-PDU

Valeur (1): signifie TRUE si Get_Master_Diag-RES-PDU

**Bit 6: Diag.Prm_Fault (Parameter Fault, défaut de paramètre)**
FALSE (0), TRUE (1)

**Bit 7: Diag.Master_Lock**
Doit être mis à zéro si Diagnoses-RES-PDU

Valeur (0): signifie FALSE si Get_Master_Diag-RES-PDU

Valeur (1): signifie TRUE si Get_Master_Diag-RES-PDU

### 5.3.2 Codage du champ Station_status_2

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: Diag.Prm_Req (Parameterisation requested, paramétrisation demandée)**
FALSE (0), TRUE (1)

**Bit 1: Diag.Stat_Diag (Static Diagnosis, diagnostic statique)**
FALSE (0), TRUE (1)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 2: DP (DP Protocol, protocole de DP)**
doit toujours être mis à 1.

**Bit 3: Diag.WD_On (Watchdog on, chien de garde actif)**
FALSE (0), TRUE (1)

**Bit 4: Diag.Freeze_Mode**
FALSE (0), TRUE (1)

**Bit 5: Diag.Sync_Mode (Synchronisation Mode, mode synchronisation)**
FALSE (0), TRUE (1)

**Bit 6: reserved (réservé)**

**Bit 7: Diag.Deactivated**
Doit être mis à zéro si Diagnoses-RES-PDU

Valeur (0): signifie FALSE si Get_Master_Diag-RES-PDU

Valeur (1): signifie TRUE si Get_Master_Diag-RES-PDU

### 5.3.3 Codage du champ Station_status_3

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 6: reserved (réservé)**

**Bit 7: Diag.Ext_Diag_Overflow (Overflow of extended Diagnosis, débordement de diagnostic étendu)**
FALSE (0), TRUE (1)

### 5.3.4 Codage du champ Diag_Master_Add

Ce champ doit être codé comme un type de données Unsigned8 avec les valeurs suivantes:

**Décimal (0-125)**
signifie que l'adresse du maître DP (Classe 1) qui a paramétré cet esclave DP.

**Décimal (255)**
signifie que l'esclave DP n'a pas été paramétré ou a subi une paramétrisation non valide.

**Décimal (126-254)**
non autorisées

### 5.3.5 Codage du champ Ident_Number

Ce champ doit être codé comme un type de données Unsigned16.

### 5.3.6 Codage du champ Header_Octet

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 5: Block_Length**
Il doit contenir la longueur du bloc de diagnostic. Le Header_Octet lui-même doit toujours être compté.

Pour le format de diagnostic de base du Device Related Diagnosis (diagnostic relatif au dispositif), les valeurs doivent être situées dans la plage décimale de 2 à 63. Pour le format de

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

diagnostic de base de l'Identifier Related Diagnosis (diagnostic relatif à l'identificateur), les valeurs doivent être situées dans la plage décimale de 2 à 32. Pour le format de diagnostic DPV1 pour le Device Related Diagnosis (diagnostic relatif au dispositif), les valeurs doivent être situées dans la plage décimale de 4 à 63.

NOTE Les diagnostics relatifs à l'identificateur (Identifier Related Diagnoses) sont limités à 32 car l'identificateur lui-même est limité à 244.

**Bits 6 et 7: Selection**
Décimal (0) signifie Device Related Diagnosis.

Décimal (1) signifie Identifier Related Diagnosis.

### 5.3.7 Codage du champ Alarm_Type

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 6: Alarm_Type**
L'Alarm_Type doit contenir les valeurs conformes au Tableau 6.

**Tableau 6 – Plage d'Alarm_Type**

| Valeur (décimale) | Signification |
|---|---|
| 0 | reserved (réservé) |
| 1 | Diagnostic_Alarm |
| 2 | Process_Alarm |
| 3 | Pull_Alarm |
| 4 | Plug_Alarm |
| 5 | Status_Alarm |
| 6 | Update_Alarm |
| 7 – 31 | reserved (réservé) |
| 32 – 126 | spécifique constructeur |
| 127 | reserved (réservé) |

**Bit 7: doit être zéro (identifiant Alarm)**

### 5.3.8 Codage du champ Status_Type

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 6: Status_Type**
Le Status_Type doit contenir des valeurs conformes au Tableau 7.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Tableau 7 – Plage des valeurs de Status_Type**

| Valeur (décimale) | Signification |
|---|---|
| 0 | reserved (réservé) |
| 1 | Status_Message |
| 2 | Modul_Status |
| 3 | DXB_Link_Status |
| 4 à 29 | reserved (réservé) |
| 30 | PrmCmdAck |
| 31 | Red_Status |
| 32 à 126 | Spécifique constructeur |
| 127 | reserved (réservé) |

**Bit 7: doit être un (identifiant le "Status")**

### 5.3.9 Codage du champ Slot_Number

Ce champ doit être codé comme un type de données Unsigned8. La valeur doit être prise dans la plage 0 à 254. La valeur 255 est réservée.

### 5.3.10 Codage du champ Alarm_Specifier

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 et 1: Alarm_Specifier**
Ce groupe de bits doit contenir le spécificateur d'alarme avec les valeurs conformes au Tableau 8.

**Tableau 8 – Alarm_Specifier**

| Valeur (décimale) | Signification | Commentaire |
|---|---|---|
| 0 | Aucune différenciation supplémentaire | |
| 1 | Erreur apparaît et Slot perturbé | la position génère une alarme en raison d'une erreur. |
| 2 | Error disparaît et Slot est correct | la position génère une alarme et indique que la position n'a pas d'autres erreurs. |
| 3 | Error disparaît et Slot reste perturbé | la position génère une alarme et indique que la position a encore d'autres erreurs. |

**Bit 2: Additional_Acknowledge**
Valeur (0): signifie qu'aucune connaissance supplémentaire n'est utilisée.

Valeur (1): signifie qu'une connaissance supplémentaire est utilisée.

**Bits 3 à 7: Sequence_Number**
La plage de valeurs doit être de 0 à 31 (décimale).

### 5.3.11 Codage du champ Status_Specifier

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bits 0 et 1: Status_Specifier**
Décimal (0) signifie pas de différenciation plus poussée.

Décimal (1) signifie que le statut apparaît.

Décimal (2) signifie que le statut disparaît.

Décimal (3) reserved (réservé)

**Bits 2 à 7: reserved (réservé)**

### 5.3.12 Codage du champ Diagnosis_User_Data

L'un des types de données suivants doit être utilisé pour chaque champ Diagnosis_User_Data:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.3.13 Codage du champ Modul_Status_Array

Le champ Modul_Status_Array est une matrice d'octets qui doit contenir au moins un et au maximum 61 octets appelés Modul_Status_Octet. Un Modul_Status_Octet doit consister en au moins un et au maximum 4 statuts de module appelés Modul_Status_Entry_1 à Modul_Status_Entry_4 codés en deux bits chacun. Par conséquent, le numéro de Modul_Status_Octet correspond au nombre de modules configurés dans le dispositif et doit être calculé comme suit:

a) $N = 1$ si $M_{le\ plus\ grand} \leq 4$

b) $N = M_{le\ plus\ grand}$ DIV 4 +1 si $M_{le\ grand\ grand}$ MOD $4 \neq 0$

c) $N = M_{le\ plus\ grand}$ DIV 4 si $M_{le\ grand\ grand}$ MOD $4 = 0$

où

N est le nombre d'octets Modul_Status_Octet ou le nombre d'éléments de la matrice,

$M_{le\ plus\ grand}$ est le plus grand numéro des modules configurés dans le dispositif (maximum 244).

Le dernier Modul_Status_Octet (octet N) peut ne pas contenir toutes les 4 entrées de statut de module. Si $M_{le\ plus\ grand}$ MOD $4 \neq 0$, l'octet N n'est pas complètement rempli et les bits restants doivent être mis à zéro et appelés "Padding Bits" (bits de bourrage).

Les statuts des modules du dispositif doivent être structurés dans l'ordre croissant sans trous. Le statut du numéro de module un est toujours dans le Modul_Status_Entry_1 du Modul_Status_Octet numéro un. Le codage d'un Modul_Status_Octet doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

NOTE Le terme "modules configurés" s'adresse aux modules physiques ou virtuels d'un dispositif qui ont eu un champ de format connexe dans Chk_Cfg-REQ-PDU. Les modules qui ne font pas partie de la configuration sont purement locaux et ne sont donc pas liés au champ Modul_Status_Array.

**Bits 0 et 1: Modul_Status_Entry_1**
Il doit contenir le statut de module d'un module configuré avec le numéro qui respecte m MOD 4 = 1. Des bits de bourrage ne doivent pas être utilisés ici.

**Bits 2 et 3: Modul_Status_Entry_2**
Il doit contenir le statut de module d'un module configuré avec le numéro qui respecte m MOD 4 = 2. Autrement, des bits de bourrage (00 binaire) doivent être utilisés.

**Bits 4 et 5: Modul_Status_Entry_3**
Il doit contenir le statut de module d'un module configuré avec le numéro qui respecte m MOD 4 = 3. Autrement, des bits de bourrage (00 binaire) doivent être utilisés.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bits 6 et 7: Modul_Status_Entry_4**
Ces 2 bits doivent contenir le statut de module d'un module configuré avec le numéro qui respecte m MOD 4 = 0. Autrement, des bits de bourrage (00 binaire) doivent être utilisés.

où
m est le numéro du module courant avec 1 ≤ m ≤ N.

La Figure 7 illustre la disposition des champs Modul_Status_Entry_(1 à 4) comme exemple d'un dispositif avec 6 modules configurés.

| | **Numéro de bit** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Modul_ Status_ Octet 1 | status module numéro 4 | | status module numéro 3 | | status module numéro 2 | | status module numéro 1 | |
| Modul_ Status_ Octet 2 | Bits de bourrage | | Bits de bourrage | | status module numéro 6 | | status module numéro 5 | |
| | Modul_Status_Entry_4 | | Modul_Status_Entry_3 | | Modul_Status_Entry_2 | | Modul_Status_Entry_1 | |

**Figure 7 – Exemple de Modul_Status_Array**

Chaque entrée Modul_Status_Entry_1 à Modul_Status_Entry_4 doit contenir le statut de module conforme au Tableau 9.

**Tableau 9 – Plage des Modul_Status_Entry (1 à 4)**

| Valeur (décimale) | Signification |
|---|---|
| 0 | données valides |
| 1 | données non valides: les données du module correspondant ne sont pas valides en raison d'une erreur (par exemple: court-circuit). |
| 2 | données non valides / mauvais module: Les données du module correspondant ne sont pas valides, en raison d'un mauvais module en place. |
| 3 | données non valides / aucun module: Les données du module correspondant ne sont pas valides, car aucun module n'est en place. |

### 5.3.14 Codage du champ Identifier_Diagnosis_Data_Array

Le champ Identifier_Diagnosis_Data_Array est une matrice d'octets qui doit contenir au moins un et au maximum 31 octets appelés Identifier_Diagnosis_Data_Octet. Un Identifier_Diagnosis_ Data_Octet doit consister en au moins un et au plus 8 entrées de diagnostic appelées Identifier_Diagnosis_Entry_1 à Identifier_Diagnosis_Entry_8. Par conséquent, le numéro de l'Identifier_Diagnosis_Data_Octet correspond au nombre de modules configurés dans le dispositif et doit être calculé comme suit:

a) N = 1 si $M_{le\ plus\ grand} \leq 8$

b) N = $M_{le\ plus\ grand}$ DIV 8 +1 si $M_{le\ plus\ grand}$ MOD 4 ≠ 0

c) N = $M_{le\ plus\ grand}$ DIV 8 si $M_{le\ plus\ grand}$ MOD 4 = 0

où

N est le nombre d'octets Identifier_Diagnosis_Data_Octet ou le nombre d'éléments de la matrice,

$M_{le\ plus\ grand}$ est le plus grand numéro des modules configurés dans le dispositif (maximum 244).

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Le dernier Identifier_Diagnosis_Data_Octet (octet N) peut ne pas contenir toutes les 8 entrées de diagnostic. Si $M_{le\ plus\ grand}$ MOD 8≠0, l'octet N n'est pas complètement rempli et les bits restants doivent être mis à zéro et appelés "Padding Bits" (bits de bourrage).

NOTE Le terme DIV signifie la division sans reste. Le terme MOD signifie le reste de la division.

Les diagnostics d'identificateur des modules du dispositif doivent être structurés dans l'ordre croissant sans trous. Le diagnostic d'identificateur du module numéro un est toujours placé dans l'Identifier_Diagnosis_Entry_1 de l'Identifier_Diagnosis_Data_Octet numéro un. Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: Identifier_Diagnosis_Entry_1**
Ce bit doit être mis à un si le module avec le numéro qui respecte m MOD 8 = 1 indique le diagnostic; autrement, il doit être mis à zéro. Un bit de bourrage ne doit pas être utilisé ici.

**Bit 1: Identifier_Diagnosis_Entry_2**
Ce bit doit être mis à un si le module avec le numéro qui respecte m MOD 8 = 2 indique le diagnostic; autrement, il doit être mis à zéro. Un bit de bourrage doit être utilisé s'il n'y aucun module configuré.

**Bit 2: Identifier_Diagnosis_Entry_3**
Ce bit doit être mis à un si le module avec le numéro qui respecte m MOD 8 = 3 indique le diagnostic; autrement, il doit être mis à zéro. Un bit de bourrage doit être utilisé s'il n'y aucun module configuré.

**Bit 3: Identifier_Diagnosis_Entry_4**
Ce bit doit être mis à un si le module avec le numéro qui respecte m MOD 8 = 4 indique le diagnostic; autrement, il doit être mis à zéro. Un bit de bourrage doit être utilisé s'il n'y aucun module configuré.

**Bit 4: Identifier_Diagnosis_Entry _5**
Ce bit doit être mis à un si le module avec le numéro qui respecte m MOD 8 = 5 indique le diagnostic; autrement, il doit être mis à zéro. Un bit de bourrage doit être utilisé s'il n'y aucun module configuré.

**Bit 5: Identifier_Diagnosis_Entry _6**
Ce bit doit être mis à un si le module avec le numéro qui respecte m MOD 8 = 6 indique le diagnostic; autrement, il doit être mis à zéro. Un bit de bourrage doit être utilisé s'il n'y aucun module configuré.

**Bit 6: Identifier_Diagnosis_Entry _7**
Ce bit doit être mis à un si le module avec le numéro qui respecte m MOD 8 = 7 indique le diagnostic; autrement, il doit être mis à zéro. Un bit de bourrage doit être utilisé s'il n'y aucun module configuré.

**Bit 7: Identifier_Diagnosis_Entry_8**
Ce bit doit être mis à un si le module avec le numéro qui respecte m MOD 8 = 0 indique le diagnostic; autrement, il doit être mis à zéro. Un bit de bourrage doit être utilisé s'il n'y aucun module configuré.

où m est le numéro du module courant avec 1 ≤ m ≤ N.

### 5.3.15 Codage du champ Identifier_Number

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 5: Identifier_Number**
Les valeurs zéro à 63 (en décimal) sont valides.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Bits 6 à 7: Selection**
**Décimal (2) signifie "Channel Related Diagnosis" (diagnostic relatif à la voie)**

### 5.3.16 Codage du champ Channel_Number

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 5: Channel_Number**
Les valeurs zéro à 63 (en décimal) sont valides.

**Bits 6 et 7: Input_Output_Selection**
Décimal (0) reserved (réservé)

Décimal (1) signifie entrée.

Décimal (2) signifie sortie.

Décimal (3) signifie entrée et sortie.

### 5.3.17 Codage du champ Type_of_Diagnosis

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 4: Error_Type**
L'Error_Type doit être positionné comme montré dans le Tableau 10.

**Tableau 10 – Type d'erreur (Error_Type)**

| Valeur (décimale) | Signification |
|---|---|
| 0 | reserved (réservé) |
| 1 | court-circuit |
| 2 | Sous-tension |
| 3 | Surtension |
| 4 | Surcharge |
| 5 | Température excessive |
| 6 | rupture de ligne |
| 7 | valeur de limite supérieure dépassée |
| 8 | valeur de limite inférieure dépassée |
| 9 | Erreur |
| 10 à 31 | Voir ChannelErrorType dans la CEI 61158-6-10, paragraphe 6.2.5.2. |

**Bits 5 à 7: Channel_Type**
Décimal (0): non spécifique, peut être utilisé pour n'importe quel type

Décimal (1): signifie bit.

Décimal (2): signifie 2 bits

Décimal (3): signifie 4 bits

Décimal (4): signifie octet.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Décimal (5): signifie mot.

Décimal (6): signifie 2 mots.

Décimal (7): reserved (réservé)

### 5.3.18 Codage du champ Revision_Number

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 5: Revision_Number**
Les valeurs un à 63 (en décimal) sont valides.

**Bit 6: doit être mis à un.**

**Bit 7: doit être mis à un.**

### 5.3.19 Codage du champ Publisher_Address

Ce champ doit être codé comme un type de données Unsigned8. La plage de valeurs doit être de zéro à 125. Ce paramètre doit contenir l'adresse de l'esclave DP comme Publisher (éditeur).

### 5.3.20 Codage du champ Publisher_Status

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 7: Link_Status**
TRUE (1) signifie "aucune erreur et actif", FALSE (0) erreur ou inactif au cours de la dernière période de surveillance.

**Bit 6: Link_Error**
TRUE (1) signifie erreur de longueur, FALSE (0) signifie aucune erreur.

**Bits 0 à 5: reserved (réservé)**

### 5.3.21 Codage du champ RedSpecifier

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 1: Status_Specifier**
Décimal (0): signifie pas de différenciation plus poussée.

**Bits 3 à 7: Seq Number**
Contient le numéro de séquence du dernier PrmCmd traité

### 5.3.22 Codage du champ Function

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: reserved (réservé)**
**Bit 1: Primary**
TRUE (1) signifie qu'une demande Primary a été exécutée en dernier, FALSE (0) signifie qu'aucune demande Primary n'a été exécutée en dernier.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 2: Start MS1**
TRUE (1) signifie qu'une demande Start MS1 a été exécutée en dernier, FALSE (0) signifie qu'aucune demande Start MS1 n'a été exécutée en dernier.

**Bit 3: Stop MS1**
TRUE (1) signifie qu'une demande Stop MS1 a été exécutée en dernier, FALSE (0) signifie qu'aucune demande Stop MS1 n'a été exécutée en dernier.

**Bit 4: Check_Properties**
TRUE (1) signifie qu'une demande Check_Properties a été exécutée en dernier, FALSE (0) signifie qu'aucune demande Check_Properties n'a été exécutée en dernier.

**Bit 5: reserved (réservé)**
**Bit 6: MasterStateClear**
TRUE (1) signifie qu'une demande MasterStateClear a été exécutée en dernier, FALSE (0) signifie qu'aucune demande MasterStateClear n'a été exécutée en dernier.

**Bit 7: reserved (réservé)**

### 5.3.23 Codage du champ Red_Status1

Ce champ représente le statut du composant qui a envoyé le statut. Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: Backup**
TRUE (1) signifie que le composant est dans l'état Backup (secours), FALSE (0) signifie que composant n'est pas dans l'état Backup.

**Bit 1: Primary**
TRUE (1) signifie que le composant est dans l'état Primary (primaire), FALSE (0) signifie que composant n'est pas dans l'état Primary.

**Bit 2: HW-Defect**
TRUE (1) signifie que le composant a un Hardware Defect (défaut de matériel), FALSE (0) signifie qu'il n'a pas de Hardware Defect

**Bit 3: DataExchange**
TRUE (1) signifie que le composant est dans l'état DataExchange (échange de données), FALSE (0) signifie que composant n'est pas dans l'état DataExchange.

**Bit 4: Fail Safe**
TRUE (1) signifie que le composant est dans l'état Fail Safe (à sécurité intégrée), FALSE (0) signifie que composant n'est pas dans l'état Fail Safe.

**Bit 5: DatarateSet**
TRUE (1) signifie que le composant a un Datarate (débit de données) mis, FALSE (0) signifie qu'il n'a pas de Datarate (débit de données) mis.

**Bit 6: TohStarted**
TRUE (1) signifie que le composant a un Toh lancé, FALSE (0) signifie qu'il n'a pas de Toh lancé

**Bit 7: reserved (réservé)**

### 5.3.24 Codage du champ Red_Status2

Ce champ représente le statut d'un composant partenaire du composant qui a envoyé le statut. Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 0: Backup**
TRUE (1) signifie que le composant est dans l'état Backup (secours), FALSE (0) signifie que composant n'est pas dans l'état Backup.

**Bit 1: Primary**
TRUE (1) signifie que le composant est dans l'état Primary (primaire), FALSE (0) signifie que composant n'est pas dans l'état Primary.

**Bit 2: HW-Defect**
TRUE (1) signifie que le composant a un Hardware Defect (défaut de matériel), FALSE (0) signifie qu'il n'a pas de Hardware Defect

**Bit 3: DataExchange**
TRUE (1) signifie que le composant est dans l'état DataExchange (échange de données), FALSE (0) signifie que composant n'est pas dans l'état DataExchange.

**Bit 4: Fail Safe**
TRUE (1) signifie que le composant est dans l'état Fail Safe (à sécurité intégrée), FALSE (0) signifie que composant n'est pas dans l'état Fail Safe.

**Bit 5: DatarateSet**
TRUE (1) signifie que le composant a un Datarate (débit de données) mis, FALSE (0) signifie qu'il n'a pas de Datarate (débit de données) mis.

**Bit 6: TohStarted**
TRUE (1) signifie que le composant a un Toh lancé, FALSE (0) signifie qu'il n'a pas de Toh lancé

**Bit 7: reserved (réservé)**

#### 5.3.25 Codage du champ Red_Status3

Ce champ doit être codé comme un type de données Unsigned8. Les valeurs sont spécifiques à un utilisateur.

### 5.4 Section de codage relative à la PDU de paramétrisation

#### 5.4.1 Codage du champ Station_status

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: reserved (réservé)**

**Bit 1: reserved (réservé)**

**Bit 2: reserved (réservé)**

**Bit 3: WD_On (Watchdog on, chien de garde activé)**
FALSE (0), TRUE (1)

**Bit 4: Freeze_Req**
Valeur (0): le mode Freeze n'est pas demandé.

Valeur (1): le mode Freeze est demandé.

**Bit 5: Sync_Req**
Valeur (0): le mode Sync n'est pas demandé.

**Bit 6: Unlock_Req**
Doit être utilisé comme décrit dans le Tableau 11.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Valeur (1): mode Sync demandé.

**Bit 7: Lock_Req**
Doit être utilisé comme décrit dans le Tableau 11.

**Tableau 11 – Spécification des bits Lock_Req et Unlock_Req**

| Bit 7 | Bit 6 | Signification |
|---|---|---|
| 0 | 0 | Le paramètre min $T_{SDR}$ peut être changé. Tous les autres paramètres restent inchangés. |
| 0 | 1 | L'esclave DP sera déverrouillé pour les autres maîtres. |
| 1 | 0 | L'esclave DP est verrouillé pour les autres maîtres, tous les paramètres sont acceptés (exception: min $T_{SDR}$ = 0) |
| 1 | 1 | L'esclave DP est déverrouillé pour les autres maîtres. |

### 5.4.2 Codage du champ WD_Fact_1

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 1 à 255

### 5.4.3 Codage du champ WD_Fact_2

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 1 à 255

### 5.4.4 Codage du champ min_$T_{SDR}$

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal (0 à max $T_{SDR}$ (voir Partie 4)) si opération DP

Décimal (0): signifie que la valeur courante reste inchangée.

### 5.4.5 Codage du champ Group_Ident

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

Bit 0: Group_1

Bit 1: Group_2

Bit 2: Group_3

Bit 3: Group_4

Bit 4: Group_5

Bit 5: Group_6

Bit 6: Group_7

Bit 7: Group_8

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Chacun des bits ci-dessus doit être mis à un si le dispositif appartient au numéro de groupe indiqué. Autrement, il doit être mis à zéro.

NOTE Un dispositif peut n'appartenir à aucun groupe (Group_Ident=décimal(0)) ou appartenir à tous les groupes (Group_Ident=décimal(255)).

### 5.4.6 Codage du champ User_Prm_Data_Element

L'un des types de données suivants doit être utilisé pour chaque champ User_Prm_Data_Element:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.4.7 Codage du champ DPV1_Status_1

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: reserved (réservé)**

**Bit 1: reserved (réservé)**

**Bit 2: WD_Base_1ms**
Valeur (0): la base de temps du chien de garde est égale à 10 millisecondes.

Valeur (1): la base de temps du chien de garde est égale à 1 ms.

**Bits 3 et 4: reserved (réservé)**

**Bit 5: Publisher_Enable**
FALSE (0), TRUE (1)

**Bit 6: Fail_Safe**
FALSE (0), TRUE (1)

**Bit 7: DPV1_Enable**
FALSE (0), TRUE (1)

### 5.4.8 Codage du champ DPV1_Status_2

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: Check_Cfg_Mode**
Valeur (0): vérification normale (restrictive)

Valeur (1): vérification spécifique à un utilisateur (plus flexible)

**Bit 1: reserved (réservé)**

**Bit 2: Enable_Update_Alarm**
FALSE (0), TRUE (1)

**Bit 3: Enable_Status_Alarm**
FALSE (0), TRUE (1)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 4: Enable_Manufacturer_Specific_Alarm**
FALSE (0), TRUE (1)

**Bit 5: Enable_Diagnostic_Alarm**
FALSE (0), TRUE (1)

**Bit 6: Enable_Process_Alarm**
FALSE (0), TRUE (1)

**Bit 7: Enable_Pull_Plug_Alarm**
FALSE (0), TRUE (1)

### 5.4.9 Codage du champ DPV1_Status_3

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 2: Alarm_Mode**
Décimal (0): 1 alarme de chaque type

Décimal (1): 2 alarmes au total

Décimal (2): 4 alarmes au total

Décimal (3): 8 alarmes au total

Décimal (4): 12 alarmes au total

Décimal (5): 16 alarmes au total

Décimal (6): 24 alarmes au total

Décimal (7): 32 alarmes au total

**Bit 3: Prm_Structure**
FALSE (0), TRUE (1)

**Bit 4: IsoM_Req**
FALSE (0), TRUE (1)

**Bits 5 et 6: reserved (réservé)**
**Bit 7: PrmCmd**
FALSE (0), TRUE (1)

### 5.4.10 Codage du champ Structure_Length

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 5 à 244, 255

Le champ Structure_Length contient le nombre d'octets, y compris lui-même. Une valeur de 255 signifie que tous les éléments du Prm_User_Data suivant cet élément appartiennent à cette structure.

### 5.4.11 Codage du champ Structure_Type

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 0 à 1: reserved (réservé)

Décimal 2: PrmCmd

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Décimal 3: DXB Linktable

Décimal 4: IsoM Parameter (paramètre IsoM)

Décimal 5: F-Parameter (paramètre F)

Décimal 6: reserved (réservé)

Décimal 7: DXB Subscribertable (table d'abonnés DXB)

Décimal 8: Time AR Parameter

Décimal 9 à 31: reserved (réservé)

Décimal 32 à 128: Spécifique constructeur

Décimal 129: User Prm Data (données Prm utilisateur)

Décimal 130 à 255: reserved (réservé)

#### 5.4.12 Codage du champ Version

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 0: reserved (réservé)

Décimal 1: Version 1

Décimal 2 à 255: reserved (réservé)

#### 5.4.13 Codage du champ Publisher_Addr

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 0 à 125, 128

Décimal 126, 127, 129 à 255: reserved (réservé)

Dans le cas de DXB-Subscribertable, la valeur 128 indique une entrée pour le maître DP (Classe 1).

#### 5.4.14 Codage du champ Publisher_Length

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 1 à 244

Décimal 0, 245 à 255: reserved (réservé)

Dans le cas d'une entrée Master Data dans DXB-Subscribertable, la valeur de ce champ n'est pas pertinente et doit être mise à 0.

#### 5.4.15 Codage du champ Sample_Offset

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 0 à 243

Décimal 244 à 255: reserved (réservé)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.4.16 Codage du champ Sample_Length

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 1 à 244

Décimal 0, 245 à 255: reserved (réservé)

#### 5.4.17 Codage du champ Dest_Slot_Number

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 1 à 244

Décimal 0, 245 à 255: reserved (réservé)

#### 5.4.18 Codage du champ Offset_Data_Area

Ce champ doit être codé comme un type de données Unsigned8 avec la plage de valeurs suivante:

Décimal 0 à 244

Décimal 245 à 255: reserved (réservé)

#### 5.4.19 Codage du champ $T_{BASE_DP}$

Ce champ doit être codé comme un type de données Unsigned32 avec les valeurs autorisées en décimal de 375, 750, 1500 (valeur par défaut), 3000, 6000, 12000. Toutes les autres valeurs sont réservées.

#### 5.4.20 Codage du champ $T_{DP}$

Ce champ doit être codé comme un type de données Unsigned16 avec la plage de valeurs de 154 à $2^{16}$-1.

#### 5.4.21 Codage du champ $T_{MAPC}$

Ce champ doit être codé comme un type de données Unsigned8.

#### 5.4.22 Codage du champ $T_{BASE_IO}$

Ce champ doit être codé comme un type de données Unsigned32 les valeurs autorisées en décimal 375, 750, 1500 (valeur par défaut), 3000, 6000, 12000. Toutes les autres valeurs sont réservées.

#### 5.4.23 Codage du champ $T_I$

Ce champ doit être codé comme un type de données Unsigned16.

#### 5.4.24 Codage du champ $T_O$

Ce champ doit être codé comme un type de données Unsigned16.

#### 5.4.25 Codage du champ $T_{DX}$

Ce champ doit être codé comme un type de données Unsigned32.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.4.26 Codage du champ $T_{PLL_W}$

Ce champ doit être codé comme un type de données Unsigned16 avec la plage de valeurs de 1 à $2^{16}$-1.

#### 5.4.27 Codage du champ $T_{PLL_D}$

Ce champ doit être codé comme un type de données Unsigned16 avec la plage de valeurs de 0 à $2^{16}$ –1.

#### 5.4.28 Codage du champ Specifier

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 et 1: Specifier** (spécificateur)
Décimal (0): signifie pas de différenciation plus poussée.

**Bits 3 à 7: Seq Number**
Contient le numéro de séquence du PrmCmd.

#### 5.4.29 Codage du champ Function

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: reserved (réservé)**
**Bit 1: Primary**
TRUE (1) signifie qu'une une demande Primary est demandée, FALSE (0) signifie qu'aucune demande Primary n'est demandée

**Bit 2: Start MS1**
TRUE (1) signifie qu'une une demande Start MS1 est demandée, FALSE (0) signifie qu'aucune demande Start MS1 n'est demandée

**Bit 3: Stop MS1**
TRUE (1) signifie qu'une une demande Stop MS1 est demandée, FALSE (0) signifie qu'aucune demande Stop MS1 n'est demandée

**Bit 4: Check_Properties**
TRUE (1) signifie qu'une demande Check_Properties est demandée, FALSE (0) signifie qu'aucune demande Check_Properties n'est demandée.

**Bit 5: reserved (réservé)**
**Bit 6: MasterStateClear**
TRUE (1) signifie que MasterState est Clear, FALSE (0) signifie que le MasterState est Operate.

**Bit 7: reserved (réservé)**

#### 5.4.30 Codage du champ Properties

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: Primary**
TRUE (1) signifie qu'une une demande Primary est utilisée, FALSE (0) signifie qu'aucune demande Primary n'est utilisée

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 1: Start Stop MS1**
TRUE (1) signifie qu'une une demande Start Stop MS1 est utilisée, FALSE (0) signifie qu'aucune demande Start Stop MS1 n'est utilisée

**Bit 2: AddressChange**
TRUE (1) signifie que Address Change (changement d'adresse) est requis avec le rôle Primary/Backup, FALSE (0) signifie qu'aucun Address Change n'est requis avec le rôle Primary/Backup

**Bit 3: AddressOffset**
TRUE (1) signifie que Address Offset (décalage d'adresse) est 64, FALSE (0) signifie que Address Offset est 0.

**Bits 4 à 7: reserved (réservé)**

#### 5.4.31 Codage du champ Output Hold Time

Ce champ doit être codé comme un type de données Unsigned16 avec la plage de valeurs de 0 à $2^{16}$-1.

#### 5.4.32 Codage du champ Clock Sync Interval

Ce champ doit être codé comme un type de données Unsigned16 avec la plage de valeurs de 0 à $2^{16}$-1.

#### 5.4.33 Codage du champ CS Delay Time

Ce champ doit être codé comme un type de données Network Time Difference.

### 5.5 Section de codage relative aux configurationPDUs

#### 5.5.1 Codage du champ Cfg_Identifier

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 3: Data_Length**
Décimal(0): signifie 1 octet ou 1 mot conforme au Length_Format

Décimal(1): signifie 2 octets ou 2 mots conformes au Length_Format

Décimal(2): signifie 3 octets ou 3 mots conformes au Length_Format

Décimal(3): signifie 4 octets ou 4 mots conformes au Length_Format

Décimal(4): signifie 5 octets ou 5 mots conformes au Length_Format

…

Décimal(15): signifie 16 octets ou 16 mots conformes au Length_Format

**Bits 4 à 5: Input_Output_Selection**
Décimal(0): ne doit pas être utilisé ici, signifie format d'identificateur spécifique

Décimal(1): signifie entrée.

Décimal(2): signifie sortie.

Décimal(3): signifie entrée et sortie.

**Bit 6: Length_Format**
Valeur (0): signifie structure d'octet utilisée

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Valeur (1): signifie structure de mot utilisée (octet de poids fort transféré le premier, gros boutiste)

**Bit 7: Consistency** (cohérence)
Valeur (0): signifie cohérence d'octet ou de mot

Valeur (1): signifie cohérence sur toute la longueur

### 5.5.2 Codage du champ Special_Cfg_Identifier

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 3: Length_of_Manufacturer_Specific_Data**
Les informations relatives à longueur des données spécifiques à un constructeur doivent être dépendantes de l'utilisation dans Chk_Cfg-REQ-PDU ou Get_Cfg-RES-PDU comme montré dans le Tableau 12 et le Tableau 13. Les champs de type de données doivent toujours être comptés, s'ils sont présents.

**Bits 4 et 5: doivent être mis à zéro (décimal)**

**Bits 6 et 7: Input_Output_Selection**
Décimal(0): signifie position vide, absence de données d'entrée ou de sortie pour ce module

Décimal(1): signifie qu'un octet de longueur pour les données d'entrée suit.

Décimal(2): signifie qu'un octet de longueur pour les données de sortie suit.

Décimal(3): signifie qu'un octet de longueur pour les données de sortie suivi d'un octet de longueur pour les données d'entrée suit.

**Tableau 12 – Plage de Length_of_Manufacturer_Specific_Data s'il est utilisé dans Chk_Cfg-REQ-PDU**

| Valeur (décimale) | Signification |
|---|---|
| 0 | Aucune donnée spécifique à un constructeur ne suit; aucune donnée dans Real_Cfg_Data. |
| 1 à 14 | Des données de longueur spécifiée spécifiques à un constructeur suivent; elles doivent être identiques aux données dans Real_Cfg_Data. |
| 15 | Aucune donnée spécifique à un constructeur ne suit; la vérification peut être omise. |

**Tableau 13 – Plage de Length_of_Manufacturer_Specific_Data s'il est utilisé dans Get_Cfg-RES-PDU**

| Valeur (décimale) | Signification |
|---|---|
| 0 | Aucune donnée spécifique à un constructeur ne suit. |
| 1 à 14 | Des données de longueur spécifiée spécifiques à un constructeur suivent; |
| 15 | Non autorisées |

### 5.5.3 Codage du champs Length_Octet

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 5: Data_Length**
Décimal(0): signifie 1 octet ou 1 mot conforme au Length_Format

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Décimal(1): signifie 2 octets ou 2 mots conformes au Length_Format

...

Décimal(63): signifie 64 octets ou 64 mots conformes au Length_Format

**Bit 6: Length_Format**
Valeur (0): signifie structure d'octet utilisée

Valeur (1): signifie structure de mot utilisée (octet de poids fort transféré le premier, gros boutiste)

**Bit 7: Consistency** (cohérence)
Valeur (0): signifie cohérence d'octet ou de mot

Valeur (1): signifie consistance sur toute la longueur indiquée dans Data_Length.

### 5.5.4 Codage du champ Manufacturer_Specific_Data

L'un des types de données suivants doit être utilisé pour chaque champ Manufacturer_Specific_Data:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.5.5 Codage du champ Extended_Length_Octet

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 5: Data_Length**
Décimal(0): signifie 1 octet.

Décimal(1): signifie 2 octets.

Décimal(2): signifie 3 octets.

Décimal(3): signifie 4 octets.

Décimal(4): signifie 5 octets.

...

Décimal(63): signifie 64 octets.

**Bit 6: Format**
Ce bit doit être mis à zéro, ce qui signifie qu'une structure d'octet est utilisée.

**Bit 7: Consistency** (cohérence)
Ce bit doit être mis à un, ce qui signifie la cohérence sur toute la longueur.

### 5.5.6 Codage du champ Data_Type

Ce champ doit être codé comme un type de données Unsigned8 avec des valeurs conformes à la colonne code dans le Tableau 14.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 14 – Valeurs (codes) pour les types de données**

| Nom de Data type | Code (décimal) | Nombre d'octets pour le type de données |
|---|---|---|
| Boolean | 1 | 1 |
| Integer8 | 2 | 1 |
| Integer16 | 3 | 2 |
| Integer32 | 4 | 4 |
| Unsigned8 | 5 | 1 |
| Unsigned16 | 6 | 2 |
| Unsigned32 | 7 | 4 |
| Floating Point 32 | 8 | 4 |
| Visible String | 9 | 1, 2, 3, … |
| Octet String | 10 | 1, 2, 3, … |
| — | 11 | reserved (réservé) |
| TimeOfDay (avec indication de Date) | 12 | 6 |
| — | 13 à 14 | reserved (réservé) |
| Float64 | 15 | 8 |
| — | 16 à 49, 51 | reserved (réservé) |
| Date | 50 | 7 |
| TimeOfDay (sans indication de Date) | 52 | 4 |
| Time Difference (avec indication de Date) | 53 | 6 |
| Time Difference (sans indication de Date) | 54 | 4 |
| Integer64 | 55 | 8 |
| Unsigned64 | 56 | 8 |
| — | 57 | reserved (réservé) |
| Network Time | 58 | 8 |
| Network Time Difference | 59 | 8 |
| — | 60 à 70 | reserved (réservé) |
| — | 71 à 100 | Spécifique à un utilisateur |
| — | 101 à 127 | reserved (réservé) |
| — | 128 à 255 | Spécifique à un utilisateur (aucun codage de types de données) |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Les séquences de types de données doivent être codées comme une liste de simples types de données. La somme des longueurs des types de données doit correspondre à la longueur des données d'entrée ou de sortie. Les types de données avec longueur variable (visible string, octet string, time of day, time difference) doivent être manipulés comme un élément simple, et non en une séquence.

Le codage spécifique à un utilisateur (valeur >=128) ne doit être utilisé dans aucune combinaison avec le codage spécifique de type (valeur <=70) dans un Chk_Cfg-REQ-PDU ou Get_Cfg-REQ-PDU.

### 5.6 Section de codage relative aux PDU de commande globale

#### 5.6.1 Codage du champ Control_Command

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: reserved (réservé)**

**Bit 1: Clear_Data**
Valeur (0): aucune commande

Valeur (1): vider la sortie (mettre les données de sortie dans un état de sécurité)

**Bit 2: UnFreeze**
Valeur (0): aucune commande

Valeur (1): annule la commande Freeze.

**Bit 3: Freeze**
Valeur (0): aucune commande

Valeur (1): geler (freeze) les valeurs d'entrée.

**Bit 4: Unsync**
Valeur (0): aucune commande

Valeur (1): annule la commande Sync.

**Bit 5: Sync**
Valeur (0): aucune commande

Valeur (1): synchroniser les valeurs de sortie.

**Bit 6: reserved (réservé)**

**Bit 7: reserved (réservé)**
Le Tableau 15 illustre la signification et la priorité des bits 2 à 5.

**Tableau 15 – Spécification des bits pour Un-/Sync et Un-/Freeze**

| Bit 2 resp. 4 | Bit 3 resp. 5 | Signification |
|---|---|---|
| 0 | 0 | aucune fonction |
| 0 | 1 | la fonction est activée. |
| 1 | 0 | la fonction est désactivée. |
| 1 | 1 | la fonction est désactivée. |

#### 5.6.2 Codage du champ Group_Select

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

Bit 0: Group_1

Bit 1: Group_2

Bit 2: Group_3

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Bit 3: Group_4

Bit 4: Group_5

Bit 5: Group_6

Bit 6: Group_7

Bit 7: Group_8

Chacun des bits ci-dessus doit être mis à un si le la commande de commandes appartient au numéro de groupe indiqué. Autrement, il doit être mis à zéro. Zéro signifie que tous les esclaves attribués doivent exécuter la Control Command (commande de commandes).

NOTE Une commande de commandes peut n'appartenir à aucun groupe (Group_Ident=décimal(0)) ou appartenir à tous les groupes (Group_Ident=décimal(255)).

### 5.7 Section de codage relative aux clock-value-PDU

#### 5.7.1 Codage du champ Clock_value_time_event

Le codage de ce champ doit être comme le type de données Network Time avec la longueur fixe de 8.

#### 5.7.2 Clock_value_previous_TE

Le codage de ce champ doit être comme le type de données Network Time avec la longueur fixe de 8.

#### 5.7.3 Codage du champ Clock_value_status1

Le codage de ce champ doit être conforme à 3.4, avec C et CV mis en correspondance avec le paramètre local time diff dans le service de temps établi. Les bits individuels doivent avoir la signification suivante:

**Bit 0: reserved (réservé)**

**Bit 1: reserved (réservé)**

**Bits 2 à 6: CV (Correction Value, valeur de correction)**
Valeur (0): 0 min

Valeur (1): 30 min

Valeur (2): 60 min

Valeur (3): 90 min

**...**

Valeur (31): 930 min

**Bit 7: C (Signe de CV)**
Valeur (0): CV à ajouter à Time

Valeur (1): CV à retrancher de Time

#### 5.7.4 Codage du champ Clock_value_status2

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 0: SYF (Synchronisation fault, défaut de synchronisation)**
Valeur (0): Clock_value_time_event est synchronisé

Valeur (1): Clock_value_time_event n'est pas synchronisé (avec d'autres horloges dans le système)

**Bit 1: reserved (réservé)**

**Bit 2: reserved (réservé)**

**Bits 3 et 4: CR (Accuracy, exactitude)**
Valeur (0): 1 ms

Valeur (1): 10 ms

Valeur (2): 100 ms

Valeur (3): 1 s

**Bit 5: reserved (réservé)**

**Bit 6: SWT (Summer/Winter-Time, Heure d'été/hiver)**
Valeur (0): Winter Time (Heure d'hiver)

Valeur (1): Summer Time (Heure d'été)

**Bit 7: ANH (Announcement Hour, heure d'annonce)**
Valeur (0): Aucun changement planifié au cours de la prochaine heure

Valeur (1): Un changement de SWT se produira au cours de la prochaine heure.

### 5.8 Section de codage relative à l'identification de fonction et aux erreurs

#### 5.8.1 Codage du champ Function_Num

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir les significations suivantes. La signification du champ complet est définie dans le Tableau 16.

**Bits 0 à 4: Function_Code / Error_Code**
Ces bits contiennent le Function_Code ou l'Error_Code correspondant au DLPDU_Selector. Le Function_Code doit être mis conformément au Tableau 16.

**Bits 5 et 6: PDU_Identifier**
Ces deux bits doivent être mis à 2 (décimal) pour indiquer une DP-PDU.

**Bit 7: DLPDU_Selector**
Valeur (0): signifie une DLPDU de réponse positive ou une DLPDU de demande.

Valeur (1): signifie erreur ou DLPDU de réponse négative.

Le champ Function_Num est transféré dans la DLPDU de demande et de réponse. Si le service est traité correctement, le bit 7 de Function_Num dans l'APDU de réponse doit être mis à zéro. En cas d'erreur, une DLPDU d'erreur est transférée. Dans cette APDU, Function_Num contient un code d'erreur et le bit 7 doit être mis à un.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 16 – Codage de Function_Code/ Function_Num**

| DLPDU_ Selector (décimal) | PDU_ Identifier (décimal) | Function_Code (décimal) | Signification | Champ complet: Function_Num (hexadécimal) |
|---|---|---|---|---|
| 0 | 2 | 0 | reserved (réservé) | 0x40 |
| 0 | 2 | 1 | Get_Master_Diag | 0x41 |
| 0 | 2 | 2 | Start_Seq | 0x42 |
| 0 | 2 | 3 | Download (téléchargement aval) | 0x43 |
| 0 | 2 | 4 | Upload (téléchargement vers l'amont) | 0x44 |
| 0 | 2 | 5 | End_Seq | 0x45 |
| 0 | 2 | 6 | Act_Para_Brct | 0x46 |
| 0 | 2 | 7 | Act_Param | 0x47 |
| 0 | 2 | 8 | Idle | 0x48 |
| 0 | 2 | 9 à 16 | reserved (réservé) | 0x49 à 0x50 |
| 0 | 2 | 17 | Data_Transport | 0x51 |
| 0 | 2 | 18 à 21 | reserved (réservé) | 0x52 à 0x55 |
| 0 | 2 | 22 | RM | 0x56 |
| 0 | 2 | 23 | Initiate (déclencher) | 0x57 |
| 0 | 2 | 24 | Abort (abandonner) | 0x58 |
| 0 | 2 | 25 | reserved (réservé) | 0x59 |
| 0 | 2 | 26 | reserved (réservé) | 0x5A |
| 0 | 2 | 27 | reserved (réservé) | 0x5B |
| 0 | 2 | 28 | Alarm_Ack | 0x5C |
| 0 | 2 | 29 | reserved (réservé) | 0x5D |
| 0 | 2 | 30 | Read (Lecture) | 0x5E |
| 0 | 2 | 31 | Write (Écriture) | 0x5F |

L'Error_Code doit être mis conformément au Tableau 17.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 17 – Codage de l'Error_Code/ Function_Num**

| DLPDU_ Selector (décimal) | PDU_ Identifier (décimal) | Error_Code (décimal) | Signification | Champ complet: Function_Num (hexadécimal) |
|---|---|---|---|---|
| 1 | 2 | 0 | reserved (réservé) | 0xC0 |
| 1 | 2 | 1 | FE | 0xC1 |
| 1 | 2 | 2 | NI | 0xC2 |
| 1 | 2 | 3 | AD | 0xC3 |
| 1 | 2 | 4 | EA | 0xC4 |
| 1 | 2 | 5 | LE | 0xC5 |
| 1 | 2 | 6 | RE | 0xC6 |
| 1 | 2 | 7 | IP | 0xC7 |
| 1 | 2 | 8 | SC | 0xC8 |
| 1 | 2 | 9 | SE | 0xC9 |
| 1 | 2 | 10 | NE | 0xCA |
| 1 | 2 | 11 | DI | 0xCB |
| 1 | 2 | 12 | NC | 0xCC |
| 1 | 2 | 13 | TO | 0xCD |
| 1 | 2 | 14 | CA | 0xCE |
| 1 | 2 | 15 à 16 | reserved (réservé) | 0xCF à 0xD0 |
| 1 | 2 | 17 | Erreur Data_Transport | 0xD1 |
| 1 | 2 | 18 à 22 | reserved (réservé) | 0xD2 à 0xD6 |
| 1 | 2 | 23 | Erreur d'Initiate | 0xD7 |
| 1 | 2 | 24 | reserved (réservé) | 0xD8 |
| 1 | 2 | 25 | reserved (réservé) | 0xD9 |
| 1 | 2 | 26 | reserved (réservé) | 0xDA |
| 1 | 2 | 27 | reserved (réservé) | 0xDB |
| 1 | 2 | 28 | Erreur d'Alarm_Ack | 0xDC |
| 1 | 2 | 29 | reserved (réservé) | 0xDD |
| 1 | 2 | 30 | Erreur de Read | 0xDE |
| 1 | 2 | 31 | Erreur de Write | 0xDF |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.8.2 Codage du champ Error_Decode

Le codage de ce champ doit être comme celui du type de données Unsigned8 et les valeurs doivent être mises conformément au Tableau 18.

**Tableau 18 – Valeurs d'Error_Decode**

| Valeur (décimale) | Signification |
|---|---|
| 0 à 127 | reserved (réservé) |
| 128 = | DPV1 |
| 129 à 253 = | reserved (réservé) |
| 254 à 255 = | PROFILE_SPECIFIC |

Le champ Error_Decode définit la signification des Octets Error_Code_x suivants. Les valeurs d'Error_Decode PROFILE_SPECIFIC indiquent que les paramètres Error_Code_x doivent être mis tels que définis dans les protocoles correspondants:

PROFILE_SPECIFIC

Error_Code_1= issu de la spécification de profil

Error_Code_2= issu de la spécification de profil

### 5.8.3 Codage du champ Error_Code_1

Si le champ Error_Decode = DPV1

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 3: Error_Code**
Les valeurs doivent être mises conformément à la colonne Error_Code dans le Tableau 19.

**Bits 4 à 7: Error_Class**
Les valeurs doivent être mises conformément à la colonne Error_Class dans le Tableau 19.

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Tableau 19 – Codage d'Error_Code_1 à DPV1**

| Error_Class (décimal) | Signification | Error_Code (décimal) |
|---|---|---|
| 0 à 9 | non spécifiée | non spécifié *) |
| 10 | application | 0 = read error (erreur de lecture)<br>1 = write error (erreur d'écriture)<br>2 = module failure (défaillance de module)<br>3 à 6 = not specified (non spécifié)<br>7 = busy (occupé)<br>8 = version conflict (conflit de version)<br>9 = feature not supported (caractéristique non prise en charge)<br>10 à 15 = User specific (spécifique à l'utilisateur) |
| 11 | accès | 0 = invalid index (indice non valide)<br>1 = write length error (erreur de longueur d'écriture)<br>2 = invalid slot (position non valide)<br>3 = type conflict (conflit de type)<br>4 = invalid area (domaine non valide)<br>5 = state conflict (conflit d'état)<br>6 = access denied (accès refusé)<br>7 = invalid range (plage non valide)<br>8 = invalid parameter (paramètre non valide)<br>9 = invalid type (type non valide)<br>10= backup (secours)<br>11 à 15 = spécifique à un utilisateur |
| 12 | ressource | 0 = read constrain conflict (conflit de contrainte en lecture)<br>1 = write constrain conflict (conflit de contrainte en écriture)<br>2 = resource busy (ressource occupée)<br>3 = resource unavailable (ressource non disponible)<br>4 à 7 = not specified (non spécifiée)<br>8 à 15 = Spécifique à un utilisateur |
| 13 à 15 | Spécifique à un utilisateur | |
| NOTE Les valeurs non spécifiées sont utilisées pour desservir d'anciens codes et sont destinées à être transmises inchangées à l'application. | | |

Si le champ Error_Decode = PROFILE_SPECIFIC

Error_Code_1= issu de la spécification de profil, a besoin d'être spécifié dans la définition de profil.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.8.4 Codage du champ Error_Code_2

Si le champ Error_Decode = DPV1

Le codage de ce champ doit être comme celui du type de données Unsigned8. Les valeurs sont spécifiques à un utilisateur.

Si le champ Error_Decode = PROFILE_SPECIFIC

Error_Code_2= issu de la spécification de profil, a besoin d'être spécifié dans la définition de profil.

## 5.9 Section de codage relative à la PDU de diagnostic de maître

### 5.9.1 Codage du champ MDiag_Identifier

Le codage de ce champ doit être comme celui du type de données Unsigned8 et les valeurs doivent être mises conformément au Tableau 20.

**Tableau 20 – Valeurs de MDiag_Identifier**

| Valeur (décimale) | Signification |
|---|---|
| 0-125 | Diag_Data de l'esclave DP |
| 126 | System_Diagnostics |
| 127 | Master_Status |
| 128 | Data_Transfer_List |
| 129-255 | reserved (réservé) |

### 5.9.2 Codage du champ System_Diagnosis

Le codage de ce champ doit être comme le type de données Octet String avec la longueur fixe de 16. Les 16 octets de System_Diagnosis doivent être codés comme suit

**Octet 1**
Le bit 0 doit être mis à un si le numéro de poste 0 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Le bit 1 doit être mis à un si le numéro de poste 1 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Le bit 2 doit être mis à un si le numéro de poste 2 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Le bit 3 doit être mis à un si le numéro de poste 3 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Le bit 4 doit être mis à un si le numéro de poste 4 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Le bit 5 doit être mis à un si le numéro de poste 5 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Le bit 6 doit être mis à un si le numéro de poste 6 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Le bit 7 doit être mis à un si le numéro de poste 7 a rapporté un diagnostic; autrement, il doit être mis à zéro.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Octet 2**
Le bit 0 doit être mis à un si le numéro de poste 8 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Le bit 1 doit être mis à un si le numéro de poste 9 a rapporté un diagnostic; autrement, il doit être mis à zéro.

etc.

**Octet 16**
.

.

Le bit 5 doit être mis à un si le numéro de poste 125 a rapporté un diagnostic; autrement, il doit être mis à zéro.

Les bits 6 et 7 doivent être mis à zéro (non utilisés).

### 5.9.3 Codage du champ USIF_State

Ce champ doit être codé comme un type de données Unsigned8. Les valeurs suivantes sont définies:

Valeur(0x40): doit être mise si le mode de fonctionnement du maître DP (Classe 1) est STOP.

Valeur(0x80): doit être mise si le mode de fonctionnement du maître DP (Classe 1) est CLEAR.

Valeur(0xC0): doit être mise si le mode de fonctionnement du maître DP (Classe 1) est OPERATE.

### 5.9.4 Codage du champ Hardware_Release_DP

Ce champ doit être codé comme un type de données Unsigned8.

### 5.9.5 Codage du champ Firmware Release_DP

Ce champ doit être codé comme un type de données Unsigned8.

### 5.9.6 Codage du champ Hardware_Release_User

Ce champ doit être codé comme un type de données Unsigned8.

### 5.9.7 Codage du champ Firmware Release_User

Ce champ doit être codé comme un type de données Unsigned8.

### 5.9.8 Codage du champ Data_Transfer_List

Le codage de ce champ doit être comme le type de données Octet String avec la longueur fixe de 16. Les 16 octets de Data_Transfer_List doivent être codés comme suit

**Octet 1**
Le bit 0 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 0; autrement, il doit être mis à zéro.

Le bit 1 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 1; autrement, il doit être mis à zéro.

Le bit 2 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 2; autrement, il doit être mis à zéro.

Le bit 3 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 3; autrement, il doit être mis à zéro.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Le bit 4 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 4; autrement, il doit être mis à zéro.

Le bit 5 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 5; autrement, il doit être mis à zéro.

Le bit 6 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 6; autrement, il doit être mis à zéro.

Le bit 7 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 7; autrement, il doit être mis à zéro.

**Octet 2**
Le bit 0 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 8; autrement, il doit être mis à zéro.

Le bit 1 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 9; autrement, il doit être mis à zéro.

etc.

**Octet 16**
.

.

Le bit 5 doit être mis à un si l'échange de données s'est accompli avec le numéro de poste 125; autrement, il doit être mis à zéro.

Les Bits 6 et 7 doivent être mis à zéro (non utilisés).

### 5.10 Section de codage relative aux PDU upload/download/act para

#### 5.10.1 Codage du champ Area_Code_UpDownload

Le codage de ce champ doit être comme celui du type de données Unsigned8 et les valeurs doivent être mises conformément au Tableau 21.

**Tableau 21 – Valeurs pour Area_Code_UpDownload**

| Valeur (décimale) | Signification |
|---|---|
| 0 à 125 | Jeu de paramètres d'esclave DP (voir le champ DP_Slave_Parameter_Set) |
| 126 | reserved (réservé) |
| 127 | Jeu de paramètres de Bus (voir le champ Bus_Parameter_Set) |
| 128 | reserved (réservé) |
| 129 | compteurs statistiques (voir le champ Statistic_Counter) |
| 130 à 135 | reserved (réservé) |
| 136 à 139 | Jeu de paramètres de maître (voir le champ Master_Parameter_Set) |
| 140 à 254 | reserved (réservé) |
| 255 | Aucune protection d'accès locale dans Start_Seq-REQ-PDU ou Start_Seq-RES-PDU |

#### 5.10.2 Codage du champ Timeout

Le codage de ce champ doit être comme celui du type de données Unsigned16 et la base de temps pour le temporisateur doit être de 1 ms.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

### 5.10.3 Codage du champ Max_Len_Data_Unit

Le codage de ce champ doit être comme celui du type de données Unsigned8 et la valeur pour Max_Len_Data_Unit doit être située dans la plage de un à 240.

NOTE La valeur minimale pour Max_Len_Data_Unit est 48 en cas d'Initiate-RES_PDU.

### 5.10.4 Codage du champ Add_Offset

Le codage de ce champ doit être comme celui du type de données Unsigned16.

### 5.10.5 Codage du champ Data

L'un des types de données suivants doit être utilisé pour chaque champ Data:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.10.6 Codage du champ Data_Len

Le codage de ce champ doit être comme celui du type de données Unsigned8 et la valeur pour Data_Len doit être située dans la plage de un à 240.

### 5.10.7 Codage du champ Area_CodeActBrct

Le codage de ce champ doit être comme celui du type de données Unsigned8 et les valeurs doivent être mises conformément au Tableau 22.

**Tableau 22 – Valeurs pour Area_CodeActBrct**

| Valeur (décimale) | Signification |
|---|---|
| 0 à 126 | reserved (réservé) |
| 127 | Bus_parameter set |
| 128 à 129 | reserved (réservé) |
| 130 à 135 | reserved (réservé) |
| 136 à 139 | Jeu de paramètres de maître |
| 140 à 254 | reserved (réservé) |
| 255 | reserved (réservé) |

### 5.10.8 Codage du champ Area_CodeAct

Le codage de ce champ doit être comme celui du type de données Unsigned8 et les valeurs doivent être mises conformément au Tableau 23.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 23 – Valeurs pour Area_CodeAct**

| Valeur (décimale) | Signification |
|---|---|
| 0 à 125 | Jeu de paramètres d'esclave DP<br>Le fanion Active dans le jeu de paramètres d'esclave DP du maître DP (Classe 1) est influencé en conséquence. L'esclave DP concerné prend part au mode d'échange cyclique de données ou est retiré de ce mode et n'est plus adressé. |
| 126 | reserved (réservé) |
| 127 | Jeu de paramètres de bus |
| 128 | mode de fonctionnement |
| 129 | reserved (réservé) |
| 130 à 135 | reserved (réservé) |
| 136 à 139 | Jeu de paramètres de maître |
| 140 à 255 | reserved (réservé) |

### 5.10.9 Codage du champ Activate

Ce champ doit être codé comme un type de données Unsigned8. Les valeurs suivantes sont définies:

Valeur(0x80) signifie activer le jeu de paramètres d'esclave DP correspondant si le champ Area_CodeAct est 0.. 125

Valeur(0x00) signifie désactiver si le champ Area_CodeAct est 0.. 125

Valeur(0xFF) signifie activer le jeu de paramètre de bus si le champ Area_CodeAct est 127.

Valeur(0x40) signifie mettre le mode de fonctionnement à STOP si le champ Area_CodeAct est 128.

Valeur(0x80) signifie mettre le mode de fonctionnement à CLEAR si le champ Area_CodeAct est 128.

Valeur(0xC0) signifie mettre le mode de fonctionnement à OPERATE si le champ Area_CodeAct est 128.

## 5.11 Section de codage relative au jeu de paramètres de bus

### 5.11.1 Codage du champ Bus_Para_Len

Ce champ doit contenir la longueur de of Bus_Para , y compris le champ Bus_Para_Len lui-même. Ce champ doit être codé comme un type de données Unsigned16. La plage de valeurs doit être de 66 à $2^{16}$-1.

### 5.11.2 Codage du champ DL_Add

Ce champ doit être codé comme un type de données Unsigned8. La plage de valeurs doit être de zéro à 125. Le paramètre doit contenir sa propre adresse du maître DP.

### 5.11.3 Codage du champ Data_rate

Ce champ doit être codé comme un type de données Unsigned8. Les valeurs doivent être mises comme montré dans le Tableau 24:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 24 – Valeurs pour Data_rate**

| Valeur (décimale) | Signification |
|---|---|
| 0 | 9,6 kbit/s |
| 1 | 19,2 kbit/s |
| 2 | 93,75 kbit/s |
| 3 | 187,5 kbit/s |
| 4 | 500 kbit/s |
| 5 | reserved (réservé) |
| 6 | 1 500 kbit/s |
| 7 | 3 000 kbit/s |
| 8 | 6 000 kbit/s |
| 9 | 12 000 kbit/s |
| 10 | 31,25 kbit/s |
| 11 | 45,45 kbit/s |
| 12 à 255 | reserved (réservé) |

### 5.11.4 Codage du champ $T_{SL}$, min $T_{SDR}$, max $T_{SDR}$

Ces paramètres sont décrits dans la partie 3. Le codage de ces champs doit être comme celui du type de données Unsigned16.

### 5.11.5 Codage du champ $T_{QUI}$, $T_{SET}$, G, HSA, max_retry_limit

Ces paramètres sont décrits dans la partie 3. Le codage de ces champs doit être comme celui du type de données Unsigned8.

### 5.11.6 Codage du champ $T_{TR}$ (Target Token Rotation time, temps de rotation cible du jeton)

Ce champ est décrit dans la partie 3. Le codage de ce champ doit être comme celui du type de données Unsigned32.

### 5.11.7 Codage du champ Bp_Flag (Busparameter flag, fanion Busparameter)

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 2: reserved (réservé)**

**Bits 3 et 4: Isochronous Mode (mode isochrone)**
Décimal (0): Non synchronisé

Décimal (1): Buffered Synchronized (tamponné synchronisé)

Décimal (2): Enhanced Synchronized (renforcé synchronisé)

Décimal (3): reserved (réservé)

**Bit 5: IsoM Sync**
Ce bit doit être mis à un si le maître DP (Classe 1) doit envoyer un Global_Control-REQ-PDU avec Control Command Sync en cas de fonctionnement isochrone. Autrement, ce bit doit être mis à zéro.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 6: IsoM Freeze**
Ce bit doit être mis à un si le maître DP (Classe 1) doit envoyer un Global_Control-REQ-PDU avec Control Command Freeze en cas de fonctionnement isochrone. Autrement, ce bit doit être mis à zéro.

**Bit 7: Error_Action_Flag**
Ce bit doit être mis à un si le maître DP (Classe 1) doit changer le mode de fonctionnement en cas d'erreur. Autrement, ce bit doit être mis à zéro.

### 5.11.8 Codage du champ Min_Slave_Interval

Le codage de ce champ doit être comme celui du type de données Unsigned16 avec la plage de valeurs de 1 à $2^{16}$-1 et la base de temps pour le temporisateur doit être de 100 µs.

### 5.11.9 Codage du champ Poll_Timeout

Le codage de ce champ doit être comme celui du type de données Unsigned16 avec la plage de valeurs de 1 à $2^{16}$-1 et la base de temps pour le temporisateur doit être de 1 ms.

### 5.11.10 Codage du champ Data_Control_Time

Le codage de ce champ doit être comme celui du type de données Unsigned16 avec la plage de valeurs de 1 à $2^{16}$-1 et la base de temps pour le temporisateur doit être de 10 ms.

### 5.11.11 Codage du champ Alarm_Max

Le codage de ce champ doit être comme celui du type de données Unsigned8 avec la plage de valeurs de 7 à 32.

### 5.11.12 Codage du champ Max_User_Global_Control

Le codage de ce champ doit être comme celui du type de données Unsigned8 avec la plage de valeurs de un à 255. La valeur Zéro est réservée.

NOTE Ce paramètre définit le nombre maximum de demandes Global_Control qui peuvent être lancées simultanément par l'Utilisateur. Le paramètre décrit l'aptitude du maître DP. Une valeur commode dans la pratique pour Max_User_Global_Control est 16. Pour chacun des 8 groupes d'esclaves, une commande SYNC et une commande FREEZE peuvent être lancées simultanément.

### 5.11.13 Codage du champ Master_User_Data_Len

Ce champ doit contenir la longueur de Master_User_Data, y compris le champ Master_User_Data_Len lui-même. Ce champ doit être codé comme un type de données Unsigned16. La plage de valeurs doit être de 2 à $2^{16}$-1.

### 5.11.14 Codage du champ Master_Class2_Name

Le codage de ce champ doit être comme le type de données Visible String avec la longueur fixe de 32.

### 5.11.15 Codage du champ Master_User_Data

L'un des types de données suivants doit être utilisé pour chaque champ Master_User_Data:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.11.16 Codage du champ $T_{CT}$

Le codage de ce champ doit être comme celui du type de données Unsigned32 avec la plage de valeurs de1 à $2^{24}$ - 1.

### 5.11.17 Codage du champ max$T_{SH}$

Le codage de ce champ doit être comme celui du type de données Unsigned8 avec la plage de valeurs de 1 à $2^{8}$-1.

## 5.12 Section de codage relative au jeu de paramètres d'esclave

### 5.12.1 Codage du champ Slave_Para_Len

Le codage de ce champ doit être comme celui du type de données Unsigned16 et la valeur pour le Slave_Para_Len doit être ans la plage de zéro à $2^{16}$ - 1. Ce paramètre doit contenir la longueur de Slave_Para, y compris le paramètre de longueur "length" lui-même. Un jeu de paramètres d'esclave DP peut être supprimé en mettant le Slave_Para_Len à zéro.

### 5.12.2 Codage du champ SI_Flag (slave flag, fanion esclave)

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: reserved (réservé)**

**Bit 1: Extra_Alarm_SAP**
Ce bit doit être mis si le maître DP (Classe 1) doit acquitter des alarmes par le biais de SAP 50. Autrement, si le maître DP (Classe 1) acquitte des alarmes par le biais de SAP 51, ce bit ne doit pas être mis.

**Bit 2: DPV1_Data_Types**
Ce bit doit être mis si des types de données étendus sont utilisés dans la configuration (Extended_Special_Identifier_Format). Autrement, si seuls les types octet ou word sont utilisés avec Identifier_Format ou Special_Identifier_Format, ce bit ne peut pas être mis.

**Bit 3: DPV1_Supported**
Ce bit doit être mis pour activer les extensions DPV1. Autrement, les extensions doivent être désactivées.

**Bit 4: Publisher_Enable**
Ce bit doit être mis pour activer la fonction d'éditeur. Autrement, la fonction d'éditeur doit être désactivée.

**Bit 5: Fail_Safe**
Ce bit doit être mis pour forcer le maître DP (Classe 1) pour acheminer une NULL-PDU au cours du mode de fonctionnement "Clear". Autrement, les données avec la valeur zéro seront envoyées au cours du mode "Clear".

**Bit 6: New_Prm**
Ce bit doit être mis pour forcer le maître DP (Classe 1) à acheminer une nouvelle Set_Prm-REQ-PDU au cours de la phase de transfert de données. Autrement, ce bit doit être mis à zéro.

**Bit 7: Actif**
Ce bit doit être mis pour forcer le maître DP (Classe 1) à activer l'esclave DP correspondant. Autrement, ce bit doit être mis à zéro.

### 5.12.3 Codage du champ Slave_Type

Ce champ doit être codé comme un type de données Unsigned8 avec des valeurs conformes au Tableau 25:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 25 – Valeurs pour Slave_Type**

| Valeur (décimale) | Signification |
|---|---|
| 0 | Esclave DP |
| 1 à 15 | reserved (réservé) |
| 16 à 255 | spécifique constructeur |

### 5.12.4 Codage du champ Max_Diag_Data_Len

Ce champ doit être codé comme un type de données Unsigned8 et avec des valeurs dans la plage de 6 à 244.

### 5.12.5 Codage du champ Max_Alarm_Len

Ce champ doit être codé comme un type de données Unsigned8 et avec des valeurs dans la plage de 4 à Min(Max_Diag_Data_Len-6, 64).

### 5.12.6 Codage du champ Max_Channel_Data_Length

Ce champ doit être codé comme un type de données Unsigned8 et avec des valeurs dans la plage de 4 à 244.

NOTE Ce paramètre définit la taille maximale d'APDU pour les esclaves DP correspondants sur la MS1-AR.

### 5.12.7 Codage du champ Diag_Upd_Delay

Ce champ doit être codé comme un type de données Unsigned8 et avec des valeurs dans la plage de 0 à 15 (extensible jusqu'à 255).

NOTE Le paramètre est utilisé pour compter le nombre de Slave_Diag.cnf dans l'état DIAG2 pendant que Diag_Data.Prm_Req est encore mis (pour les esclaves avec rendement réduit).

### 5.12.8 Codage du champ Alarm_Mode

Ce paramètre spécifie le nombre maximal d'alarmes actives possibles.

Ce champ doit être codé comme un type de données Unsigned8 avec des valeurs conformes au Tableau 26:

**Tableau 26 – Valeurs pour Alarm_Mode**

| Valeur (décimale) | Signification |
|---|---|
| 0 | 1 alarme de chaque type |
| 1 | 2 alarmes au total |
| 2 | 4 alarmes au total |
| 3 | 8 alarmes au total |
| 4 | 12 alarmes au total |
| 5 | 16 alarmes au total |
| 6 | 24 alarmes au total |
| 7 | 32 alarmes au total |

### 5.12.9 Codage du champ Add_Sl_Flag

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 0: NA_To_Abort**
Ce bit doit être mis si le maître DP (Classe 1) doit poursuivre l'échange de données au cas où l'esclave DP correspondant ne répondrait pas. Autrement, ce bit ne doit pas être mis.

**Bit 1: Ignore_ACIr**
Ce bit doit être mis si le maître DP (Classe 1) doit exclure du comportement "auto-clear" (autoeffacement) l'esclave DP correspondant. Autrement, ce bit ne doit pas être mis.

**Bits 2 à 7: reserved (réservé)**

### 5.12.10 Codage du champ MS1_Timeout

Ce champ doit être codé comme un type de données Unsigned16 et avec des valeurs dans la plage décimale 1 à $2^{16}$-1. La valeur 0 est réservée, mais peut être utilisée pour des valeurs de secours.

### 5.12.11 Codage du champ Prm_Data_Len

Ce paramètre contient la longueur de Prm_Data, y compris le paramètre de longueur "length".

Ce champ doit être codé comme un type de données Unsigned16 et avec des valeurs dans la plage de 9 à 246.

### 5.12.12 Codage du champ Prm_Data

Ce champ doit être codé comme un Set_Prm-REQ-PDU.

### 5.12.13 Codage du champ Cfg_Data_Len

Ce paramètre contient la longueur de Cfg_Data, y compris le paramètre de longueur "length".

Ce champ doit être codé comme un type de données Unsigned16 et avec des valeurs dans la plage de 3 à 246.

### 5.12.14 Codage du champ Cfg_Data

Ce champ doit être codé comme un Chk_Cfg-REQ-PDU.

### 5.12.15 Codage du champ Add_Tab_Len

Ce paramètre contient la longueur de Add_Tab, y compris le paramètre de longueur "length".

Ce champ doit être codé comme un type de données Unsigned16 et avec des valeurs dans la plage de 2 à $2^{16}$-31.

### 5.12.16 Codage du champ Number_of_Entries

Ce champ contient le nombre d'entrées dans la table d'adresses.

Ce champ doit être codé comme un type de données Unsigned16 et avec des valeurs dans la plage de 1 à 488.

NOTE Ce champ contient le nombre d'entrées dans la table d'affectation d'adresses dans l'hôte à l'esclave DP adressé. Au maximum 488 entrées par esclave DP sont autorisées (un maximum de 244 chaînes de configurations doublé pour les entrées et les sorties est possible).

### 5.12.17 Codage du champ Add_Tab_Entry_Header

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Bit 0: IO_Selection**
Ce bit doit être mis si les adresses suivantes appartiennent à la partie de sortie. Autrement, ce bit ne doit pas être et les adresses suivantes appartiennent à la partie d'entrée.

**Bit 1: Address_Format_Selection**
Ce bit doit être mis à un si les adresses suivantes sont codées comme Unsigned32. Ce bit doit être mis à zéro si les adresses suivantes sont codées comme Unsigned16.

**Bits 2 à 7: reserved (réservé)**

### 5.12.18 Codage du champ I/O_Data_Length

Ce champ doit être codé comme un type de données Unsigned8 et avec des valeurs dans la plage de 1 à 244.

### 5.12.19 Codage du champ I/O_Config_Address

Ce champ représente l'adresse locale de l'identificateur de configuration correspondant.

Ce champ doit être codé comme un type de données Unsigned16 si le bit Address_Format_Selection dans le champ Add_Tab_Entry_Header est mis à zéro.

Ce champ doit être codé comme un type de données Unsigned32 si le bit Address_Format_Selection dans le champ Add_Tab_Entry_Header est mis à un.

### 5.12.20 Codage du champ Host_Address

Ce champ représente l'adresse locale des données d'entrée ou de sortie correspondantes.

Ce champ doit être codé comme un type de données Unsigned16 si le bit Address_Format_Selection dans le champ Add_Tab_Entry_Header est mis à zéro.

Ce champ doit être codé comme un type de données Unsigned32 si le bit Address_Format_Selection dans le champ Add_Tab_Entry_Header est mis à un.

### 5.12.21 Codage du champ Slave_User_Data_Len

Ce paramètre contient la longueur de Slave_User_Data, y compris le paramètre de longueur "length".

Ce champ doit être codé comme un type de données Unsigned16 et avec des valeurs dans la plage de 2 à $2^{16}$-31.

### 5.12.22 Codage du champ Slave_User_Data

L'un des types de données suivants doit être utilisé pour chaque champ Slave_User_Data:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.12.23 Codage du champ Ext_Prm_Data_Len

Ce paramètre contient la longueur de Ext_Prm_Data, y compris le paramètre de longueur "length" (2 octets).

Ce champ doit être codé comme un type de données Unsigned16 et avec des valeurs dans la plage de 2 (aucun Ext_Prm_Data ne suit) et de 7 à 246.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.12.24 Codage du champ Ext_Prm_Data

Ce champ doit être codé comme un Set_Ext_Prm-REQ-PDU.

### 5.13 Section de codage relative aux compteurs statistiques

#### 5.13.1 Codage du champ DLPDU_sent_count and SD_count

Le codage de ce champ doit être comme celui du type de données Unsigned32.

#### 5.13.2 Codage du champ Error_count and SD_error_count

Le codage de ce champ doit être comme celui du type de données Unsigned16.

### 5.14 Section de codage relative à la PDU d'établissement d'adresse esclave

#### 5.14.1 Codage du champ New_Slave_Add

Ce champ doit être codé comme un type de données Unsigned8 avec des valeurs dans la plage de 0 à 125.

#### 5.14.2 Codage du champ No_Add_Change

Ce champ doit être codé comme un type de données Boolean. La valeur TRUE doit être utilisée pour indiquer que le changement de l'adresse est possible une seule fois. La valeur FALSE doit être utilisée pour indiquer que le changement de l'adresse est possible plus d'une fois.

#### 5.14.3 Codage du champ Rem_Slave_Data

L'un des types de données suivants doit être utilisé pour chaque champ Rem_Slave _Data:

Boolean, Integer, Unsigned, Floating Point, Visible String, Octet String, Date, TimeOfDay, Time-Difference.

### 5.15 Section de codage relative aux PDU initiate/abort

#### 5.15.1 Codage du champ Features_Supported_1

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: Read_Write**
Ce bit doit être mis pour indiquer que les services Read et Write sur MS2-AR sont pris en charge.

NOTE Read et Write sont obligatoires sur MS2-AR.

**Bits 1 à 7: reserved (réservé)**

#### 5.15.2 Codage du champ Features_Supported_2

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

Bits 0 à 7: reserved (réservé)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.15.3 Codage du champ Profile_Features_Supported_1

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

Bits 0 à 7: définis par le profil.

Ce champ doit être mis à zéro (décimal) si aucun profil n'est utilisé. Autrement, la valeur doit être prise dans la spécification de profil appropriée.

### 5.15.4 Codage du champ Profile_Features_Supported_2

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

Bits 0 à 7: définis par le profil.

Ce champ doit être mis à zéro (décimal) si aucun profil n'est utilisé. Autrement, la valeur doit être prise dans la spécification de profil appropriée.

### 5.15.5 Codage du champ Profile_Ident_Number

Le codage de ce champ doit être comme celui du type de données Unsigned16. La valeur zéro doit être mise si aucun profil n'est utilisé. Autrement, la valeur doit être prise dans la spécification de profil appropriée.

### 5.15.6 Codage du champ S_Type (source type, type de source)

Le codage de ce champ doit être comme celui du type de données Unsigned8. La valeur zéro doit être mise pour indiquer l'utilisation du format d'adresse courte. La valeur un doit être mise pour indiquer l'utilisation du format d'adresse longue. Les autres valeurs sont réservées.

NOTE Les termes source et destination appartiennent toujours au sens de déplacement de la PDU utilisée.

### 5.15.7 Codage du champ D_Type (destination type, type de destination)

Le codage de ce champ doit être comme celui du type de données Unsigned8. La valeur zéro doit être mise pour indiquer l'utilisation du format d'adresse courte. La valeur un doit être mise pour indiquer l'utilisation du format d'adresse longue. Les autres valeurs sont réservées.

### 5.15.8 Codage du champ S_Len (source length, longueur de source)

Ce champ doit être codé comme un type de données Unsigned8.

### 5.15.9 Codage du champ D_Len (destination length, longueur de destination)

Ce champ doit être codé comme un type de données Unsigned8.

### 5.15.10 Codage du champ S_API (source application identifier, identificateur d'application source)

Ce champ doit être codé comme un type de données Unsigned8.

### 5.15.11 Codage du champ D_API (destination application identifier, identificateur d'application destination)

Ce champ doit être codé comme un type de données Unsigned8.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.15.12 Codage du champ S_SCL (source security level, niveau de sécurité de source)

Ce champ doit être codé comme un type de données Unsigned8. la valeur zéro doit être mise pour indiquer qu'aucun niveau d'accès n'est utilisé.

### 5.15.13 Codage du champ D_SCL (destination security level, niveau de sécurité de destination)

Ce champ doit être codé comme un type de données Unsigned8. La valeur zéro doit être mise pour indiquer qu'aucun niveau d'accès n'est utilisé.

### 5.15.14 Codage du champ S_Network_Address

Le champ doit être seulement présent si le champ S_Type est mis à un.

Le codage de ce champ doit être comme le type de données Octet String avec la longueur fixe de 6.

### 5.15.15 Codage du champ D_Network_Address

Le champ doit être seulement présent si le champ D_Type est mis à un.

Le codage de ce champ doit être comme le type de données Octet String avec la longueur fixe de 6.

### 5.15.16 Codage du champ S_MAC_Address

Le champ doit être seulement présent si le champ S_Type est mis à un.

Le codage de ce champ doit être comme celui du type de données Octet String.

### 5.15.17 Codage du champ D_MAC_Address

Le champ doit être seulement présent si le champ D_Type est mis à un.

Le codage de ce champ doit être comme celui du type de données Octet String.

### 5.15.18 Codage du champ Send_Timeout

Ce champ doit être codé comme un type de données Unsigned16. La base de temps pour le temporisateur doit être de 10 ms. Les valeurs doivent être mises à partir de la plage de 1 à $2^{16}$-1.

### 5.15.19 Codage du champ Server_SAP

Ce champ doit être codé comme un type de données Unsigned8. Les valeurs doivent être mises à partir de la plage de zéro à 48.

### 5.15.20 Codage du champ Subnet

Le codage de ce champ doit être comme celui du type de données Unsigned8 et les valeurs doivent être mises conformément au Tableau 27.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 27 – Valeurs pour Subnet**

| Valeur (décimale) | Signification |
|---|---|
| 0 | signifie aucun sous-réseau |
| 1 | signifie sous-réseau local |
| 2 | signifie sous-réseau distant |
| 3 à 255 | reserved (réservé) |

### 5.15.21 Codage du champ Instance_Reason_Code

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bits 0 à 3: Reason Code** (Code de cause)
Les valeurs doivent être mises conformément au Tableau 28 si l'instance est mise à DLL.

**Tableau 28 – Valeur de code de cause si l'instance est DLL**

| Valeur (décimale) | Signification |
|---|---|
| 1 | UE (pour sa signification, voir la Partie 3) |
| 2 | RR (pour sa signification, voir la Partie 3) |
| 3 | RS (pour sa signification, voir la Partie 3) |
| 9 | NR (pour sa signification, voir la Partie 3) |
| 10 | DH (pour sa signification, voir la Partie 3) |
| 11 | LR (pour sa signification, voir la Partie 3) |
| 12 | RDL (pour sa signification, voir la Partie 3) |
| 13 | RDH (pour sa signification, voir la Partie 3) |
| 14 | DS (pour sa signification, voir la Partie 3) |
| 15 | NA (pour sa signification, voir la Partie 3) |

Les valeurs doivent être mises conformément au Tableau 29 si l'instance est mise à MS2.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Tableau 29 – Valeur de code de cause si l'instance est MS2**

| Valeur (décimale) | Signification |
|---|---|
| 1 | ABT_SE signifie erreur de séquence |
| 2 | ABT_FE signifie APDU de demande non valide reçue |
| 3 | ABT_TO signifie connexion temporisée |
| 4 | ABT_RE signifie APDU de réponse non valide reçue |
| 5 | ABT_IV signifie service non valide issu de l'Utilisateur |
| 6 | ABT_STO signifie que la valeur demandée de Send_Timeout était trop courte. |
| 7 | ABT_IA signifie informations d'adresses complémentaires non valides |
| 8 | ABT_OC signifie S-Timer a expiré, l'APDU de réponse n'a pas encore été envoyée. |

**Bits 4 et 5: Instance**
Les valeurs doivent être mises comme suit:

Décimal (0): DLL

Décimal (1): MS2

Décimal (2): USER

Décimal (3): reserved (réservé)

**Bits 6 et 7: reserved (réservé)**

### 5.16 Section de codage relative aux PDU read/write/data transport

#### 5.16.1 Codage du champ Index

Ce champ doit être codé comme un type de données Unsigned8 avec des valeurs dans la plage de 0 à 255.

#### 5.16.2 Codage du champ Length

Ce champ contient le nombre d'octets du champ Data.

Ce champ doit être codé comme un type de données Unsigned8 avec des valeurs dans la plage 0 à 240.

### 5.17 Section de codage relative aux PDU région de charge et invocation de fonction

#### 5.17.1 Codage du champ Extended_Function_Num

Outre le champ Function_Num, tel que décrit en 5.8.1, ce champ spécifie la fonction du service Load Region (région de charge) ou Function Invocation (invocation de fonction).

Le codage de ce champ doit être comme celui du type de données Unsigned8 et les valeurs doivent être mises conformément au Tableau 30.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Tableau 30 – Valeurs d'Extended_Function_Num**

| Valeur (décimale) | Signification |
|---|---|
| 0 | Initiate_Load |
| 1 | Push (Pousser) |
| 2 | Pull |
| 3 | Terminate (Mettre fin) |
| 4 | Start (Démarrer) |
| 5 | Stop (Arrêter) |
| 6 | Resume (Reprendre) |
| 7 | Reset (Réinitialiser) |
| 8 | Call |
| 9 | Get_FI_State |
| 10 à 255 | reserved (réservé) |

### 5.17.2 Codage du champ Options

Le codage de ce champ doit être conforme à 3.4 et les bits pris séparément doivent avoir la signification suivante:

**Bit 0: More_follows**
Ce bit est seulement valide pour l'APDU Push et l'APDU Pull. Ce bit doit être mis si la séquence Load Region n'est pas finie et des APDU Push ou des APDU Pull suivent. Ce bit doit être réinitialisé si la séquence Load Region est finie et aucune APDU Push ou APDU Pull ne suit.

Pour l'APDU Initiate_Load APDU, ce bit doit être réinitialisé.

**Bits 1 à 6: reserved (réservé)**

**Bit 7: Pull / Push**
Ce bit est seulement valide dans l'APDU Initiate_Load. Ce bit doit être mis, si la séquence Pull LR est initialisée. Ce bit doit être réinitialisé, si une séquence Push LR est initialisée.

Pour l'APDU Push et l'APDU Pull, ce bit doit être réinitialisé.

### 5.17.3 Codage du champ Sequence_Number

Ce champ doit être codé comme un type de données Unsigned32.

### 5.17.4 Codage du champ LR_Data

Ce champ doit être codé comme un type de données Octet String.

### 5.17.5 Codage du champ Max_Segment_Length

Ce champ doit être codé comme un type de données Unsigned8.

### 5.17.6 Codage du champ LR_Index

Ce champ doit être codé comme un type de données Unsigned16.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 5.17.7 Codage du champ LR_Length

Ce champ doit être codé comme un type de données Unsigned32.

### 5.17.8 Codage du champ Max_Response_Delay

Ce champ doit être codé comme un type de données Unsigned16.

### 5.17.9 Codage du champ Intersegment_Request_Timeout

Ce champ doit être codé comme un type de données Unsigned16.

### 5.17.10 Codage du champ User_Specific

Ce champ contient des informations spécifiques à un utilisateur. Par exemple: le format de LR_Data.

Ce champ doit être codé comme un type de données Octet String.

### 5.17.11 Codage du champ FI_Index

Ce champ doit être codé comme un type de données Unsigned16.

Valeurs permises: La place d'indices disponible est structurée comme suit:

- Indices réservés pour une utilisation spécifique au constructeur (0 - 32 767)
- Indices réservés aux services du système Profibus (32 768 - 64 999)
- Indices réservés aux services Identification et Maintenance (I&M) (65 000 - 65 199).

### 5.17.12 Codage du champ Entity Number

Ce champ doit être codé comme un type de données Unsigned8.

Valeurs permises: La signification de ce paramètre est spécifique à un profil. S'il n'est pas utilisé, il doit être mis à 0.

### 5.17.13 Codage du champ Execution_Argument

Ce champ doit être codé comme un type de données Octet String.

### 5.17.14 Codage du champ Result_Argument

Ce champ doit être codé comme un type de données Octet String.

### 5.17.15 Codage du champ FI_State

Ce champ contient l'état de l'entité Function Invocation.

Le codage de ce champ doit être comme celui du type de données Unsigned8 et les valeurs doivent être mises conformément au Tableau 31.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 31 – Valeurs de FI_State**

| Valeur (décimale) | Signification |
|---|---|
| 0 | IDLE |
| 1 | STARTING |
| 2 | RUNNING |
| 3 | STOPPING |
| 4 | STOPPED |
| 5 | RESUMING |
| 6 | UNRUNNABLE |
| 7 | RESETTING |
| 8 | WAIT-SET-IN-USE |
| 9 | WAIT-SET-IN-USE-2 |
| 10 | WAIT-DEL-IN-USE |
| 11 | WAIT-DEL-IN-USE-2 |
| 12 | EXEC-ERR |
| 13 | EXEC-TERM |
| 14 à 255 | reserved (réservé) |

Ce champ doit être codé comme un type de données OctetString avec 10 octets. La valeur doit être mise à "manufacturer specific" (spécifique à un constructeur) et doit être remplie avec 0x00 si elle est plus courte que 16.

### 5.17.16 Codage du champ IMData_Execution_Argument

#### 5.17.16.1 Généralités

Ce champ contient l'IMData_Execution_Argument du CALL-REQ-PDU. Le Codage du champ est spécifié pour:

— IM_Date dans la CEI 61158-6-10, 6.2.4.83.15,

— IM_Descriptor dans la CEI 61158-6-10, 6.2.4.83.16,

— IM_Header en5.17.16.2,

— I&M_Manufacturer_Specific_x en 5.17.16.5,

— IM_Profile_Specific_Data en 5.17.16.4,

— IM_Signature dans la CEI 61158-6-10, 6.2.4.83.17,

— IM_Tag_Function dans la CEI 61158-6-10, 6.2.4.83.13, et

— IM_Tag_Location dans la CEI 61158-6-10, 6.2.4.83.14.

#### 5.17.16.2 Codage du champ IM_Header

Ce champ doit être codé comme un type de données OctetString avec 10 octets. La valeur doit être mise à "manufacturer specific" (spécifique à un constructeur) et doit être remplie avec 0x00 si elle est plus courte que 16.

#### 5.17.16.3 Codage du champ IM_Manufacturer_ID

Ce champ doit être codé comme un type de données Unsigned16.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 5.17.16.4 Codage du champ IM_Profile_Specific_Data

Ce champ doit être codé comme un type de données OctetString avec 54 octets. La valeur doit être mise à "profile specific" (spécifique à un profil) et doit être remplie avec 0x00 si elle est plus courte que 54.

#### 5.17.16.5 Codage du champ I&M_Manufacturer_Specific_x

Ce champ doit être codé comme un type de données OctetString avec 64 octets. La valeur doit être mise à "manufacturer specific" (spécifique à un constructeur) et doit être remplie avec 0x00 si elle est plus courte que 64.

### 5.17.17 Codage du champ IMData_Result_Argument

Ce champ contient l'IMData_Result_Argument du CALL-RES-PDU. Le codage du champ est spécifié pour:

- IM_Date dans la CEI 61158-6-10, 6.2.4.83.15,
- IM_Descriptor dans la CEI 61158-6-10, 6.2.4.83.16,
- IM_Hardware_Revision dans la CEI 61158-6-10, 6.2.4.83.3,
- IM_Header en 5.17.16.2,
- IM_Manufacturer_ID en 5.17.16.3,
- I&M_Manufacturer_Specific_x en 5.17.16.5,
- IM_Profile_ID dans la CEI 61158-6-10, 6.2.4.83.8,
- IM_Profile_Specific_Data en 5.17.16.4,
- IM_Profile_Specific_Type dans la CEI 61158-6-10, 6.2.4.83.9,
- IM_Revision_Counter dans la CEI 61158-6-10, 6.2.4.83.7,
- IM_Serial_Number 6.2.4.83.2,
- IM_Signature dans la CEI 61158-6-10, 6.2.4.83.17,
- IM_Supported dans la CEI 61158-6-10, 6.2.4.83.12,
- IM_SWRevision_Bug_Fix dans la CEI 61158-6-10, 6.2.4.83.5,
- IM_SWRevision_Functional_Enhancement dans la CEI 61158-6-10, 6.2.4.83.4,
- IM_SWRevision_Internal_Change dans la CEI 61158-6-10, 6.2.4.83.6,
- IM_Tag_Function dans la CEI 61158-6-10, 6.2.4.83.13,
- IM_Tag_Location dans la CEI 61158-6-10, 6.2.4.83.14,
- IM_Version_Major dans la CEI 61158-6-10, 6.2.4.83.10,
- IM_Version_Minor dans la CEI 61158-6-10, 6.2.4.83.11,
- OrderID dans la CEI 61158-6-10, 4.8.2.2.2, et
- SWRevisionPrefix dans la CEI 61158-6-10, 4.8.2.2.4.

## 5.18 Exemples des diagnosis-RES-PDU

La Figure 8 illustre un exemple pour une APDU de diagnostic avec un message de statut relatif à un dispositif, une alarme de processus relatif à un dispositif et un diagnostic relatif à un identificateur.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | Headeroctet: diagnostic relatif à un dispositif |
| 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Status_Type: Status_Message |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Slot_Number: 2 (capteur A) |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | *Specifier* (c'est-à-dire: spécificateur):aucune différenciation supplémentaire |
| 5 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Diagnostic_User_Data: 5 (température moyenne) |
| 6 | | | | | | | | | Diagnostic_User_Data: valeur de température |
| 7 | | | | | | | | | (Unsigned16) |
| 8 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | Headeroctet: diagnostic relatif à un dispositif |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Alarm_Type: Process_Alarm |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Slot_Number: 3 (soupape B) |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | *Specifier* (c'est-à-dire: spécificateur): alarme apparaît |
| 12 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | Diagnostic_User_Data: 0x50 (limite supérieure dépassée, en pression) |
| 13 | | | | | | | | | Diagnostic_User_Data: marqueur temporel |
| 14 | | | | | | | | | (Time_of_day = 4 octets) |
| 15 | | | | | | | | | |
| 16 | | | | | | | | | |
| 17 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | Headeroctet: diagnostic relatif à l'identificateur |
| 18 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1er numéro d'identificateur avec diagnostic |
| | msb | | | | | | | | |

**Figure 8 – Exemple d'Ext_Diag_Data dans le cas du format de diagnostic DPV1 avec PDU d'alarme et de statut**

**Partie textuelle correspondante**
attributions de texte pour le capteur A et la soupape B

Unit_Diag_Area = 24-27

Valeur(1) = “Température minimale”

Valeur(2) = “Température maximale”

Valeur(5) = “Température moyenne”

Unit_Diag_Area_End

Unit_Diag_Area = 28-31

Valeur(1) = “Limite inférieure dépassée, en pression”

Valeur(5) = “Limite supérieure dépassée, en pression”

Unit_Diag_Area_End

Unit_Diag_Area = 8-15

Valeur(2) = “capteur A”

Valeur(3) = “soupape B”

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

Unit_Diag_Area_End

Unit_Diag_Area = 16-17

Valeur(1) = “alarme/statut apparaissant”

Valeur(2) = “alarme/statut disparaissant”

Unit_Diag_Area_End

Comme ces définitions sont utilisées aussi bien pour les alarmes que pour les messages de statut, il convient que leurs valeurs soient différentes. Cela signifie qu'il convient que des valeurs différentes pour les alarmes et les messages de statut soient utilisées à la même position dans le champ de diagnostic.

La Figure 9 illustre un exemple pour une APDU de diagnostic avec un diagnostic spécifique à l'utilisateur et relatif à un dispositif, un diagnostic relatif à un identificateur et un diagnostic relatif à une voie.

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets<br>1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | diagnostic relatif à un dispositif: |
| 2 | champ de diagnostic | | | | | | | | la signification des bits est définie |
| 3 | de longueur 3 | | | | | | | | spécifique à un constructeur |
| 4 | spécifique à un dispositif | | | | | | | | |
| 5 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | diagnostic relatif à l'identificateur |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | l'identificateur numéro 0 a un diagnostic. |
| 7 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | l'identificateur numéro 12 a un diagnostic. |
| 8 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | l'identificateur numéro 18 a un diagnostic. |
| 9 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | diagnostic relatif à une voie:<br>identificateur numéro 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | voie 2 |
| 11 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | surcharge, bit de voie organisé |
| 12 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | identificateur numéro 12 |
| 13 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | voie 6 |
| 14 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | valeur de limite supérieure dépassée, mot de voie organisé |
| | msb | | | | | | | | |

**Figure 9 – Exemple d'Ext_Diag_Data dans le cas du format de diagnostic de base**

### 5.19 Exemple de Chk_Cfg-REQ-PDU

La Figure 10 illustre un exemple pour une APDU Chk_Cfg avec un format d'identificateur spécial.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets<br>1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | entrée/sortie, 3 octets de données spécifiques au constructeur |
| 2 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | cohérence, sortie, 16 mots |
| 3 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | cohérence, entrée, 8 mots |
| 4 | données spécifiques au constructeur | | | | | | | | |
| 5 | | | | | | | | | |
| 6 | | | | | | | | | |
| | msb | | | | | | | | |

**Figure 10 – Exemple de format d'identificateur spécial**

### 5.20 Exemples de Chk_Cfg-REQ-PDU avec types de données DPV1

Les types de données sont codés dans la partie spécifique au constructeur.

Les simples types de données sont codés dans un octet qui contient le code du type de donnée.

Les séquences de types de données sont codées comme une liste de simples types de données. La somme des longueurs des types de données doit correspondre à la longueur des données d'entrée ou de sortie. Les types de données avec longueur variable (visible string, octet string, time of day, time difference) peuvent être manipulés comme un élément simple, et non en une séquence. Des commentaires peuvent être ajoutés à la fin de la liste des types de données.

La Figure 11 montre le format d'identificateur spécial pour un bloc d'entrées analogiques contenant une valeur (Floating Point), un Statut (Unsigned8) et des Commentaires (3 Octets).

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets<br>1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | Format spécifique au constructeur |
| 2 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Cohérence 5 Octets (Entrée) |
| 3 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Type de données "Floating Point" |
| 4 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Type de données "Unsigned8" |
| 5 | Données spécifiques au constructeur | | | | | | | | Commentaire |
| 6 | Données spécifiques au constructeur | | | | | | | | Commentaire |
| 7 | Données spécifiques au constructeur | | | | | | | | Commentaire |
| | msb | | | | | | | | |

**Figure 11 – Exemple de format d'identificateur spécial avec types de données**

La Figure 12 montre le format d'identificateur spécial pour un bloc d'entrées analogiques où la valeur de sortie peut être relue en différé.

La représentation de ce bloc de sorties analogiques est structurée comme suit:

Valeur de sortie (Floating Point)
Statut de la sortie(Unsigned8)
Valeur relue en différé (Floating Point)
Statut relu en différé (Unsigned8)
Commentaire (1 octet)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | Format spécifique au constructeur |
| 2 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Cohérence 5 Octets (Sortie) |
| 3 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Cohérence, 5 Octets (Entrée) |
| 4 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Type de données "Floating Point" |
| 5 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Type de données "Unsigned8" |
| 6 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Type de données "Floating Point" |
| 7 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Type de données "Unsigned8" |
| 8 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Commentaire |
| | msb | | | | | | | | |

**Figure 12 – Exemple de format d'identificateur spécial avec types de données**

La Figure 13 montre le format d'identificateur spécial pour une position vide avec un commentaire de 3 octets.

La représentation de cette position vide est structurée comme suit:

position vide

commentaire (3 octets)

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Format spécifique au constructeur |
| 2 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Commentaire |
| 3 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | Commentaire |
| 4 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | Commentaire |
| | msb | | | | | | | | |

**Figure 13 – Exemple de position vide avec types de données**

La Figure 14 montre l'utilisation du format d'identificateur spécial pour un dispositif à plusieurs variables qui utilise le codage spécifique à un utilisateur au lieu de types de données dans le champ data type.

La représentation de ce dispositif à plusieurs variables est structurée comme suit:

Entrée analogique (AI, Analog Input)

Sortie discrète (DO, Discrete Output) avec point de consigne (SP_D) et relecture différée (READBACK_D)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| bits | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Dispositif à plusieurs variables, AI et DO |
|---|---|---|---|---|---|---|---|---|---|
| octets<br>1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | **Entrée analogique**<br>Special Config Identifier pour l'AI en exemple (0x42) |
| 2 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Extended Length Octet pour l'entrée (0x84) |
| 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Spécifique à un utilisateur au lieu du type de données >128 (par exemple: code de blocs pour AI 0x81) |
| 4 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | Spécifique à un utilisateur (par exemple: Identification de paramètre cyclique pour AI 0x81) |
| 5 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | **Sortie discrète**<br>Special Config Identifier pour la DO en exemple (0xC2) |
| 6 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Extended Length Octet pour la sortie (0x81) |
| 7 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Extended Length Octet pour l'entrée (0x81) |
| 8 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Spécifique à un utilisateur, au lieu du type de données >128 (par exemple: code de blocs pour DO 0x84) |
| 9 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Spécifique à un utilisateur, au lieu de type de données >128 (par exemple: Identification de paramètre cyclique pour DO avec SP_D et READBACK_D 0x83) |
| | msb | | | | | | | | |

**Figure 14 – Exemple de dispositif à plusieurs variables avec des blocs de fonction AI et DO**

### 5.21 Exemple de structure de Data_Unit pour Data_Exchange

La structure de la Data_Unit pour le service Data_Exchange est définie par les identificateurs qui sont transmis à l'esclave DP avec le service Chk_Cfg à la configuration.

L'exemple suivant illustre comment les données dans la Data_Unit seront transférées, aux identificateurs donnés. La Figure 15 montre les données qui doivent être envoyées par Data_Exchange et leur ordre tel que défini par le service Chk_Cfg.

| 1. ID | 2. ID | 3. ID | 4. ID | 5. ID | 6. ID | 7. ID | |
|---|---|---|---|---|---|---|---|
| 1 B_I | 1 W_I | 2 W_O | 2 B_IO | 1 B_O | 2 W_I | 3 W_A | |

**Figure 15 – Identificateurs (ID)**

La Figure 16 montre le type et la longueur des données correspondant à leurs identificateurs et le sens de leur transfert, c'est-à-dire si elles sont entrées ou sorties.

1 B_I: 1 Octet d'entrée;
1 B_O: 1 Octet de sortie;
2 B_IO: 2 Octets d'entrée et de sortie
1 W_I: 1 mot d'entrée
1 W_O: 1 mot de sortie
etc.

**Figure 16 – Liste d'identificateurs**

La séquence des données dans la Data_Unit est montrée à la Figure 17.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Data_Exchange.req**

| Bit | msb 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | |
| 1 | Word_high | | | | (3. ID) | | | |
| 2 | Word_low | | | | (3. ID) | | | |
| 3 | Données | | | | (4. ID) | | | |
| 4 | Données | | | | (4. ID) | | | |
| 5 | Données | | | | (5. ID) | | | |
| 6 | 1.Word_high | | | | (7. ID) | | | |
| 7 | 1.Word_low | | | | (7. ID) | | | |
| 8 | 2.Word_high | | | | (7. ID) | | | |
| 9 | 2.Word_low | | | | (7. ID) | | | |
| 10 | 3.Word_high | | | | (7. ID) | | | |
| 11 | 3.Word_low | | | | (7. ID) | | | |
| 12 | etc. | | | | | | | |
| … | | | | | | | | |
| n | | | | | | | | |

**Data_Exchange.res**

| Bit | msb 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| octets | | | | | | | | |
| 1 | Données | | | | (1. ID) | | | |
| 2 | Word_high | | | | (2. ID) | | | |
| 3 | Word_low | | | | (2. ID) | | | |
| 4 | Données | | | | (4. ID) | | | |
| 5 | Données | | | | (4. ID) | | | |
| 6 | 1.Word_high | | | | (6. ID) | | | |
| 7 | 1.Word_low | | | | (6. ID) | | | |
| 8 | 2.Word_high | | | | (6. ID) | | | |
| 9 | 2.Word_low | | | | (6. ID) | | | |
| 10 | etc. | | | | | | | |
| 11 | | | | | | | | |
| 12 | | | | | | | | |
| … | | | | | | | | |
| n | | | | | | | | |

**Figure 17 – Structure de la Data_Unit pour DLPDU de demande et DLPDU de réponse**

Pour l'identificateur de configuration "empty slot" (position vide), aucune donnée ne sera transférée dans la Data_Unit.

# 6 Diagrammes d'états de protocole FAL

## 6.1 Structure globale

### 6.1.1 Fieldbus Service Protocol Machines (FSPM)

Les Diagrammes d'états FSPM (Fieldbus Service Protocol Machines «diagrammes protocolaires des services de bus de terrain») coordonnent les diagrammes d'états sous-jacents utilisés pour le traitement des divers services et relations d'applications.

Il y a un diagramme d'états pour chaque type de dispositif (FSPMS pour esclave DP, FSPMM1 pour maître DP (Classe 1) et FSPMM2 pour maître DP (Classe 2)).

### 6.1.2 Cyclique Maître vers Esclave (MS0)

Les Diagrammes d'états MSCY1 sont chargés du transfert cyclique de données entre le maître DP (Classe 1) et un esclave DP (MSCY1M/MSCY1S). Cela comprend la paramétrisation, la configuration, le diagnostic et l'échange de données ainsi que les commandes de commandes globales (Global Control). Il y a un MSCY1S pour un esclave DP et jusqu'à 125 MSCY1M pour un maître DP (Classe 1). Pour s'abonner à des objets de données d'entrée, un esclave DP utilise jusqu'à 125 SSCY1S.

### 6.1.3 Acyclique Maître (classe 1) vers Esclave (MS1)

Les Diagrammes d'états MSAC1 (MSAC1M, MSAC1S) sont chargés du transfert acyclique de données entre le maître DP (Classe 1) et un esclave DP. La relation de communications MS1 est couplée à la relation et communications cycliques et traite les services Read, Write and Alarm Ack ainsi que les services Load Region et Function Invocation.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

En outre, le diagramme d'états MSAL1M est utilisé pour gérer les alarmes au niveau du maître DP.

Il y a un MSAC1S pour un esclave DP et jusqu'à 125 MSAC1M/MSAL1M pour un maître DP (Classe 1).

### 6.1.4 Acyclique Maître (classe 2) vers Esclave (MS2)

Les Diagrammes d'états MS2 (MSAC2M, MSRM2S, MSAC2S) sont chargés du transfert acyclique de données entre le maître DP (Classe 2) et un esclave DP. La relation de communications MS2 traite les services Read, Write et Data_Transport ainsi que les services Load Region et Function Invocation.

Le gestionnaire de ressources Maître Esclave (MSRM2S «Master Slave Resource Manager») est chargé de l'attribution d'un DLSAP et des sources connexes à chaque relation de communications MSAC2 établie dans l'esclave DP. Le Master Slave Resource Manager lance en outre le diagramme d'états MSAC2S approprié.

Au maître DP (Classe 2), il existe un nombre arbitraire de diagrammes d'états MSAC2M. Le nombre maximal de diagrammes d'états MSAC2M est donné par le nombre maximal d'esclaves DP (=125), chaque esclave DP fournissant le nombre maximal de CR MS2 (=49).

### 6.1.5 Synchronisation d'horloge Maître à Esclave (MS3)

Les diagrammes d'états MSCS1 permettent d'accomplir la synchronisation des horloges entre le maître DP (Classe 1) et les esclaves DP.

Il y a tout au plus un diagramme d'états au niveau du maître DP (Classe 1) et un au niveau de l'esclave DP.

### 6.1.6 Acyclique Maître-Maître (MM1/MM2)

Les diagrammes d'états MMAC (Master Master acyclic) permettent de transférer le Master_Diag, le Master Parameter Set, le Slave Parameter Set, le Bus Parameter Set entre le maître DP (Classe 1) et le maître DP (Classe 2).

Il y a tout au plus un diagramme d'états au niveau du maître DP (Classe 1) et un au niveau du maître DP (Classe 2).

### 6.1.7 Diagramme d'état de protocole de mapping DLL (DMPM)

Les DL Mapping Protocol Machines (DMPM) connectent les autres diagrammes d'états et la couche 2. Un DMPM assure la coordination de tous les diagrammes d'états concernant la configuration et la gestion d'erreurs de l'utilisation de la couche liaison de données. Le DMPM met les fonctions en correspondance avec la DLL et les services DLM de couche 2. Le DMPM génère les nécessaires paramètres de couche 2 du service, reçoit les confirmations et indications provenant de la couche 2 et les transmet à l'utilisateur approprié du DMPM.

Il y a un diagramme d'états pour chaque type de dispositif (FSPMS pour esclave DP, FSPMM1 pour maître DP (Classe 1) et FSPMM2 pour maître DP (Classe 2)).

Interface entre DMPM et Couche 2

Le DMPM est placé directement sur l'interface Utilisateur de DLL – DLL et sur l'interface Utilisateur de DLMS – DLM comme décrit dans les définitions des services de couche liaison de données (se référer à la définition des services DL dans la CEI 61158-3-3).

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

Pour la mise en correspondance des fonctions DMPM sur la couche 2, le DMPM utilise les services suivants:

a) Services de DLL
   Send and Request Data with Reply (SRD)
   Send Data with No Acknowledge (SDN)
   Send and Request Data with Multicast Reply (MSRD)
   Clock Synchronization (CS)
b) Services DLM
   Reset
   Set Value
   Get Value
   Event
   SAP Activate
   SAP Activate Responder
   SAP Deactivate

### 6.2 Affectation des diagrammes d'états à des dispositifs

Le Tableau 32 montre l'affectation de diagrammes d'états à des dispositifs. Certains diagrammes d'états sont toujours requis et sont donc repérés par la lettre 'M' (obligatoire), les autres sont repérés par 'O' (facultatif). Les Remarques montrent les options dans les diagrammes obligatoires.

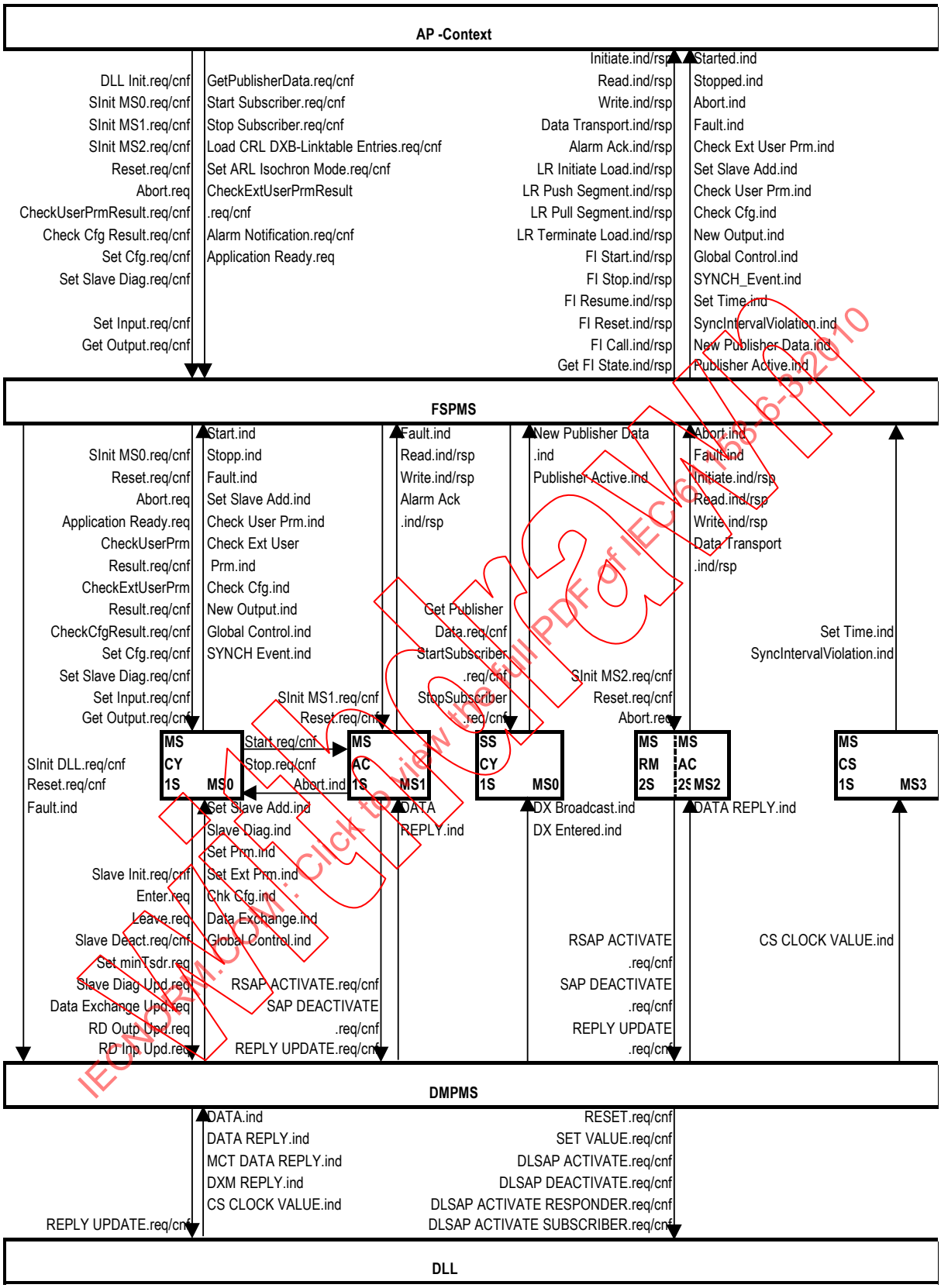
**Tableau 32 – Affectation de diagrammes d'états**

| Type du dispositif | Diagramme | Obligatoire/ Facultatif | Remarques |
|---|---|---|---|
| Esclave DP | FSPMS | M | La gestion des alarmes est facultative. |
| | MSCY1S | M | |
| | SSCY1S | O | |
| | MSAC1S | O | |
| | MSAC2S | O | |
| | MSRM2S | O | |
| | DMPMS | M | |
| Maître DP (Classe 1) | FSPMM1 | M | |
| | MSCY1M | M | |
| | MSAL1M | O | |
| | MSAC1M | O | |
| | MMAC1 | O | |
| | DMPMM1 | M | |
| Maître DP (Classe 2) | FSPMM2 | M | |
| | MSAC2M | O | |
| | MMAC2 | O | |
| | DMPMM2 | M | |

### 6.3 Vue d'ensemble d'un esclave DP

La Figure 18 illustre la structure générale de l'AL d'un esclave DP en montrant ses diagrammes d'états et les services qu'ils utilisent.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

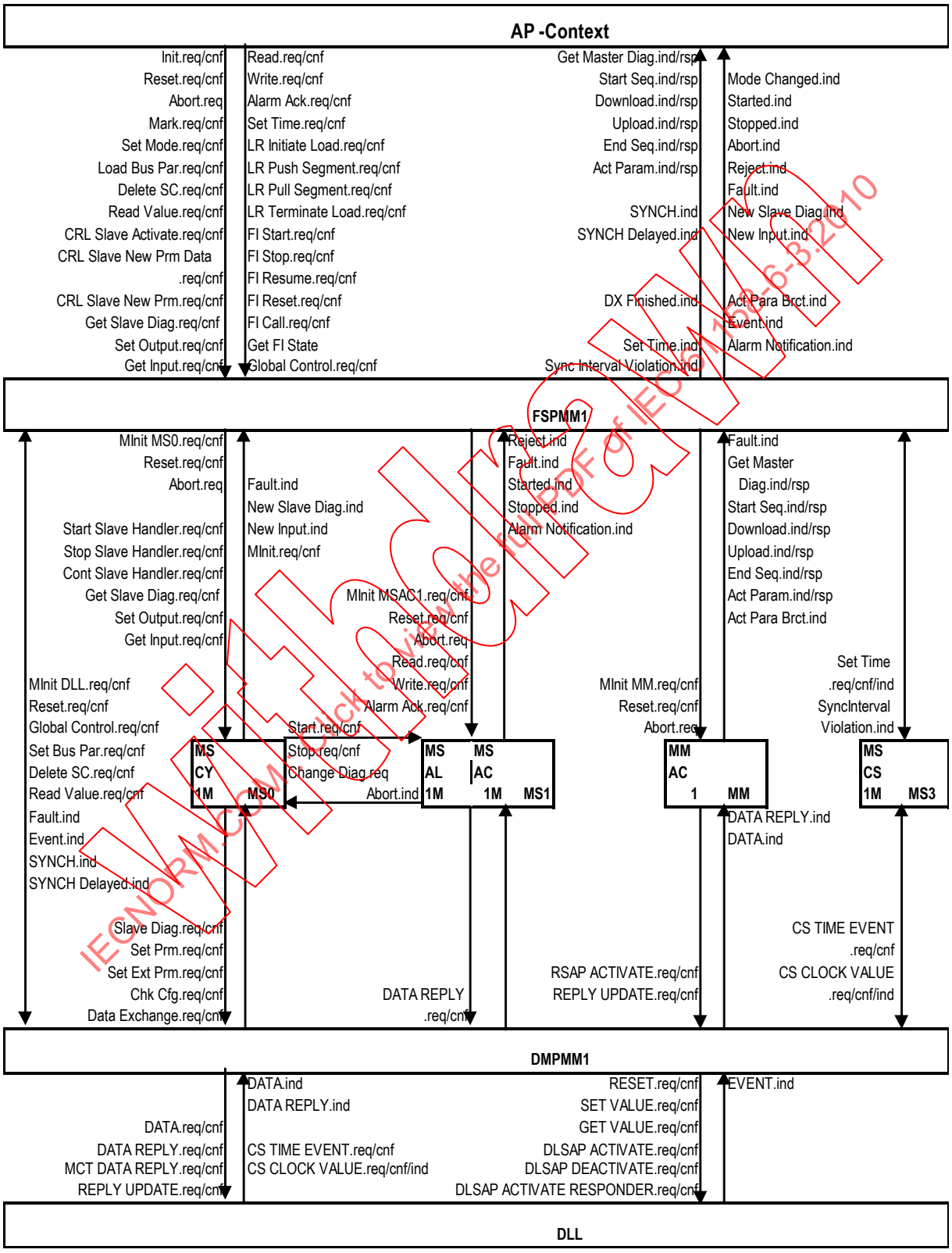



IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Figure 18 – Structuration des diagrammes protocolaires et des couches adjacentes dans un esclave DP**

### 6.4 Vue d'ensemble de maître DP (Classe 1)

La Figure 19 illustre la structure générale de l'AL d'un maître DP (Classe 1) en montrant ses diagrammes d'états et les services qu'ils utilisent.

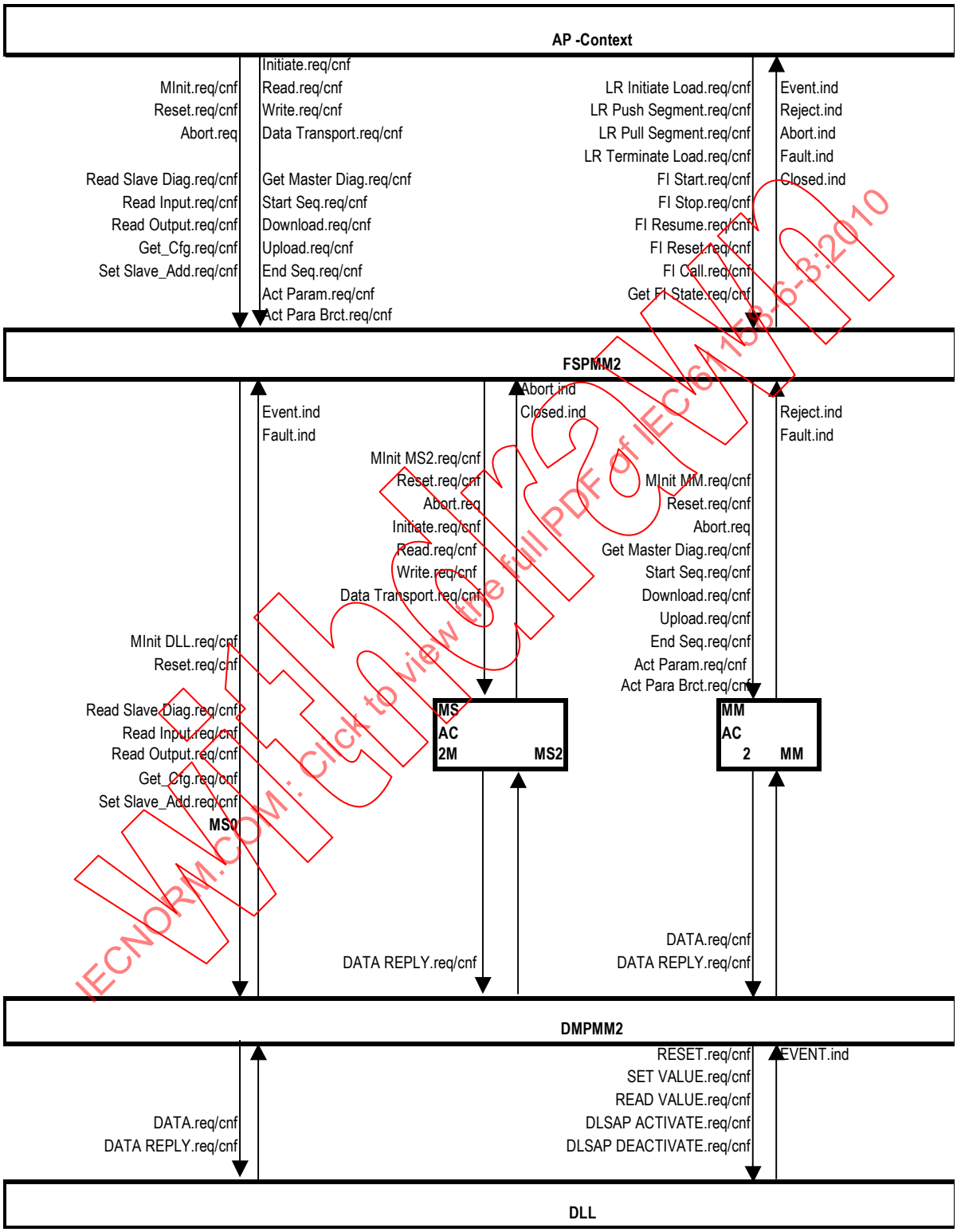



IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Figure 19 – Structuration des diagrammes protocolaires et des couches adjacentes dans un maître DP (Classe 1)**

### 6.5 Vue d'ensemble de maître DP (Classe 2)

La Figure 20 illustre la structure générale de l'AL d'un maître DP (Classe 2) en montrant ses diagrammes d'états et les services utilisés.




IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Figure 20 – Structuration des diagrammes protocolaires et des couches adjacentes dans un maître DP (Classe 2)**

### 6.6 Communication cyclique entre maître DP (Classe 1) et esclave DP

La communication entre maître DP et esclave DP est établie avec la séquence montrée à la Figure 21:

Si le maître DP souhaite communiquer avec un esclave DP, le maître DP demande les Diag_Data de l'esclave DP, afin de vérifier que l'esclave DP est prêt pour l'opération.

Cette demande de diagnostic sera répétée jusqu'à ce que l'esclave DP réponde avec les données demandées.

Si l'esclave DP répond avec les informations de diagnostic demandées, le maître DP vérifie si un autre maître DP occupe cet esclave DP. Si tel n'est pas le cas, l'esclave DP sera paramétré et configuré (Set_Prm / Chk_Cfg).

Ensuite, le maître DP demande de nouveau les données de diagnostic à l'esclave DP. Les messages suivants peuvent se produire:

- Erreur de paramétrisation ou de configuration
- L'esclave DP est occupé avec un autre maître DP.
- Diagnostic utilisateur statique, aucun transfert normal de données possible (Par exemple: Procédure de démarrage pour DP exécutée correctement, mais une certaine paramétrisation externe est en cours)
- L'esclave DP n'est pas prêt pour l'opération.

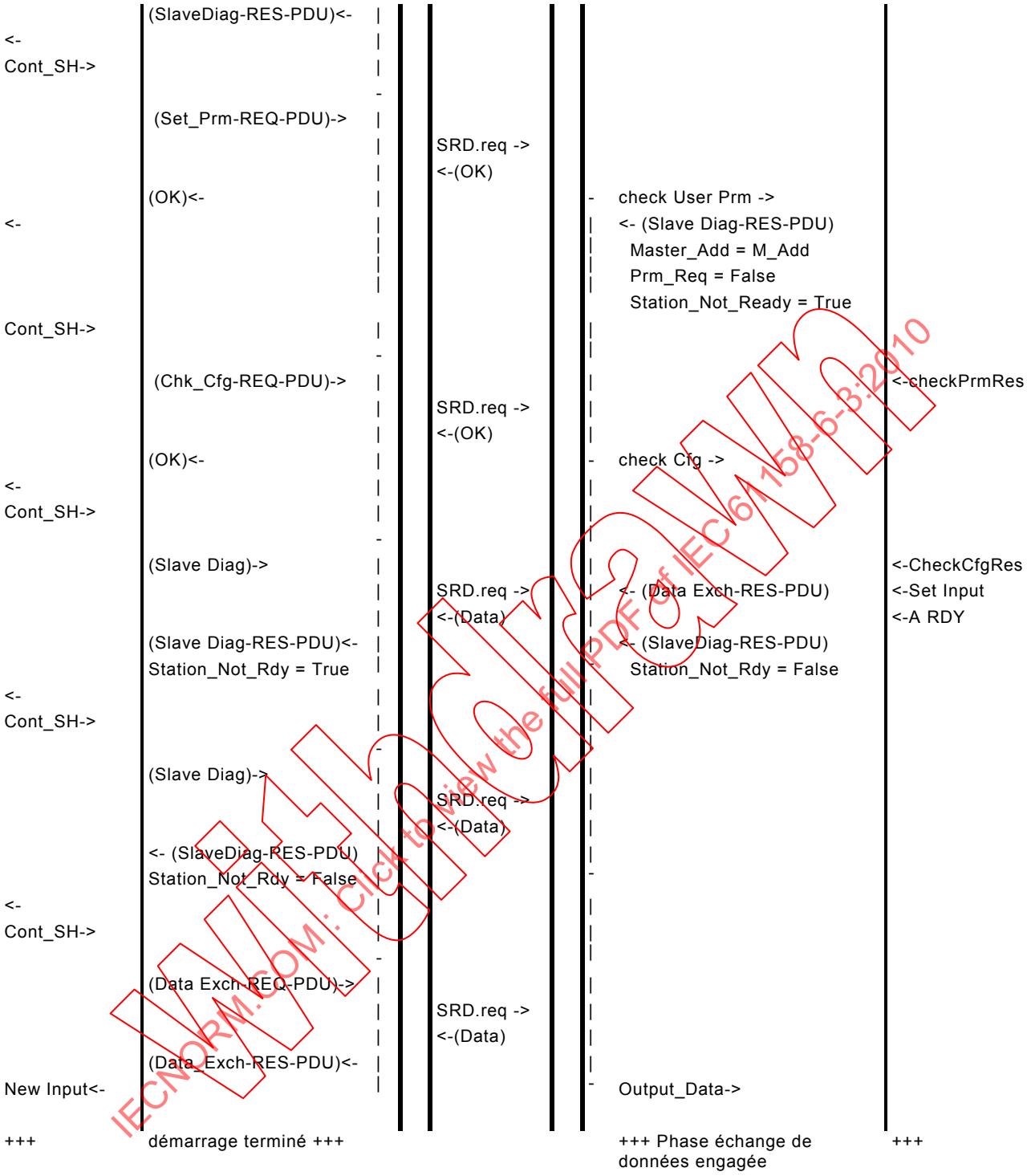
Dans les cas a) et b), le maître DP retourne à l'état de départ et vérifie de nouveau que l'esclave DP est prêt pour l'opération.

Dans les cas c) et d), le maître DP demande les informations de diagnostic provenant de l'esclave DP jusqu'à ce que les messages mentionnés disparaissent.

Si aucun message d'erreur n'apparaît, le maître DP démarre l'échange de données à destination de l'esclave DP. Les données d'entrée et de sortie seront transférées à destination et en provenance de l'esclave DP. Le maître DP (Classe 1) indique pour la première fois son propre mode de fonctionnement aux esclaves DP.

| Maître DP (Classe 1) | | | | | Esclave DP | | |
|---|---|---|---|---|---|---|---|
| | MSCY1M | MSInt | DL | | DL | WD | MSCY1S |
| | /*Démarrage du Maître*/ | | | | | | /*Esclave hors ligne*/ |
| Start_SH-> (SH=Slave Handler, descripteur d'esclave) | | - | | | | | |
| | (Slave Diag)-> | \| | | | | | <-SET_VALUE (Initialise DLL) |
| | | \| | | SRD.req -> | | | |
| | (NA)<- | \| | | | | | <-MS0_Init |
| <- | | \| | | | | | <-RSAP_ACTIVATE |
| | | \| | | | | | |
| Cont_SH-> | | \| | | | | | <- (Slave Diag-RES-PDU) Master_Add = invalid |
| | | - | | | | | |
| | (Slave Diag)-> | \| | | | | | -> |
| | | \| | | SRD.req -> | | | |
| | | \| | | <-(Data) | | | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn




IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Figure 21 – Séquence de la communication entre maître DP et esclave DP**

### 6.7 Communication acyclique entre maître DP (Classe 2) et maître DP (Classe 1)

La communication entre maître DP (Classe 2) et maître DP (Classe 1) est gérée d'une manière fixe (Request-Poll-Response «demande-sondage-réponse»), voir Figure 22, avec une exception.

Cette exception est la fonction Act_Para_Brct, qui est transférée comme une multidiffusion du maître DP (Classe 2) vers le maître DP (Classe 1) correspondant.

La communication est toujours lancée par le maître DP (Classe 2). Le maître DP (Classe 2) envoie une demande au maître DP (Classe 1) et attend la réponse. Le service est traité par le

MMAC2 du maître DP (Classe 2). Après réception de la réponse, le maître DP (Classe 2) peut faire une nouvelle demande.

À un instant donné, le maître DP (Classe 1) peut seulement communiquer avec un seul maître DP (Classe 2).

La communication du maître DP est déclenchée par un xxx.req provenant du maître DP (Classe 2). Le MMAC1 transmet cette demande à la couche 2 et vérifie le statut après réception de la réponse de DMPM. Si le statut de la réponse est NR (pas de données de réponse), il continuera avec un nouveau SRD sans L_sdu pour obtenir des données de réponse. Cette procédure sera répétée par le maître DP (Classe 2) jusqu'à

l'apparition d'une réponse erronée,

ou la réception d'une SRD_RES_PDU avec des données de réponse,

ou l'expiration du temporisateur (Timer) T1.

Le résultat sera alors transmis à l'utilisateur du maître DP (Classe 2).

Le temporisateur T1 sera lancé après réception du xxx.req dans le MMAC2 du maître DP (Classe 2) et sera arrêté après émission du xxx.cnf. La valeur chargée dans le temporisateur T1 dépend du paramètre DLL et de la performance du maître DP (Classe 1).

Le répondeur (maître DP (Classe 1)) attend une demande après une réinitialisation. Si le MMAC1 du maître DP (Classe 1) reçoit une demande valide, cette demande est transmise à l'utilisateur par le xxx.ind correspondant. À cet instant, une protection d'accès était établie au point d'accès au service local (SAP) pour la communication avec le maître DP (Classe 2). Ceci est nécessaire pour s'assurer que les données de réponse seront envoyées au bon maître DP (Classe 2).

Après réception de la réponse provenant de l'utilisateur (xxx.rsp), les données de réponse seront transmises avec un REPLY_UPDATE.req au DMPM. En outre, un temporisateur sera lancé et commandera la demande pour la réponse. Si le temporisateur expire et les données de réponse ne sont toujours pas récupérées par le maître DP (Classe 2), le MMAC1 supprime les données de réponse, car le MMAC1 présume que le maître DP (Classe 2) n'est plus disponible. Dans ce cas, la protection d'accès pour le maître DP (Classe 2) sera annulée. Il en sera également ainsi dans le cas d'une seule transaction si un DATA_REPLY.ind indique que les données de réponse sont recherchées par le maître DP (Classe 2). La valeur chargée dans le temporisateur du maître DP (Classe 1) dépend du paramètre DL et de la performance du maître DP (Classe 2).

Dans le cas d'une communication Maître-Maître, une séquence de demandes peut être mise entre parenthèses. Par conséquent, la protection d'accès sera établie par le “Start_Seq” et libérée par l'“End_Seq”. Afin d'éviter les erreurs, le temps entre deux demandes sera surveillé. La valeur pour la surveillance du temps est ajoutée à la demande Start_Seq. Avec le Start_Seq.req, l'utilisateur peut, en plus, envoyer un code de zone. Avec ce code de zone au début d'une séquence, une zone peut être réservée au niveau de l'utilisateur dans le maître DP (Classe 1).

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

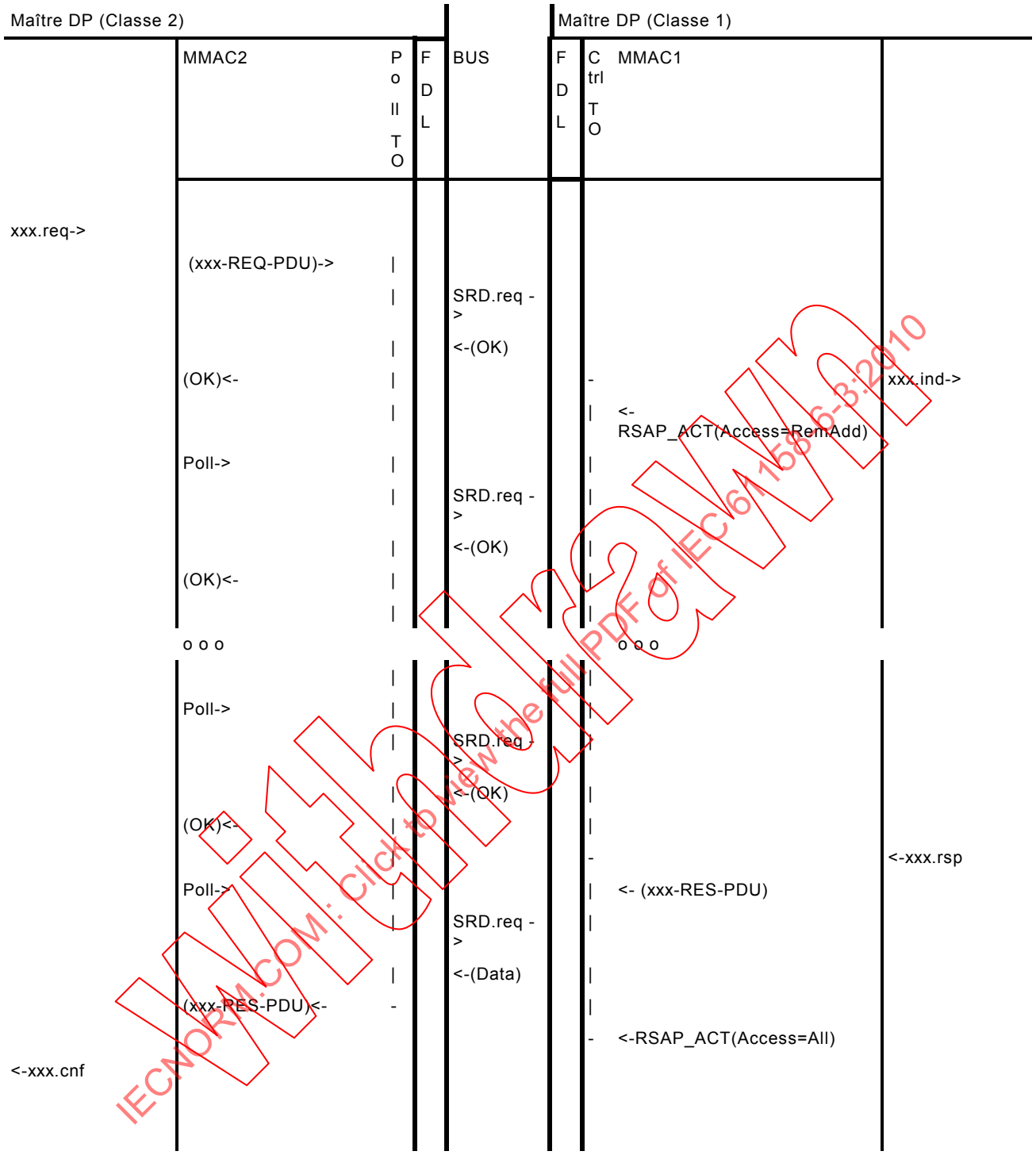



**Figure 22 – Séquence de communication entre maître DP (classe 2) et maître DP (classe 1)**

### 6.8 Communication acyclique entre maître DP (Classe 1) et esclave DP

La communication acyclique entre maître DP (Classe 1) et esclave DP est gérée d'une manière fixe (Request-Poll-Response «demande-sondage-réponse») sur la relation d'applications MS1 (voir Figure 23).

La communication est toujours lancée par le maître DP (Classe 1). La connexion est établie si l'AR de la communication cyclique MS0 est entrée dans le mode d'échange de données. Le maître DP (Classe 1) émet une demande et effectue un sondage pour une réponse.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

La communication est déclenchée par un xxx.req provenant du maître DP (Classe 1). Le MSAC1M transmet cette demande à la couche 2 et vérifie le statut après réception de la réponse de DMPMM1. Si le statut de la réponse est NR (pas de données de réponse), il continuera avec un nouveau SRD avec une NULL-PDU pour obtenir des données de réponse. Cette procédure sera répétée par le maître DP (Classe 1) jusqu'à

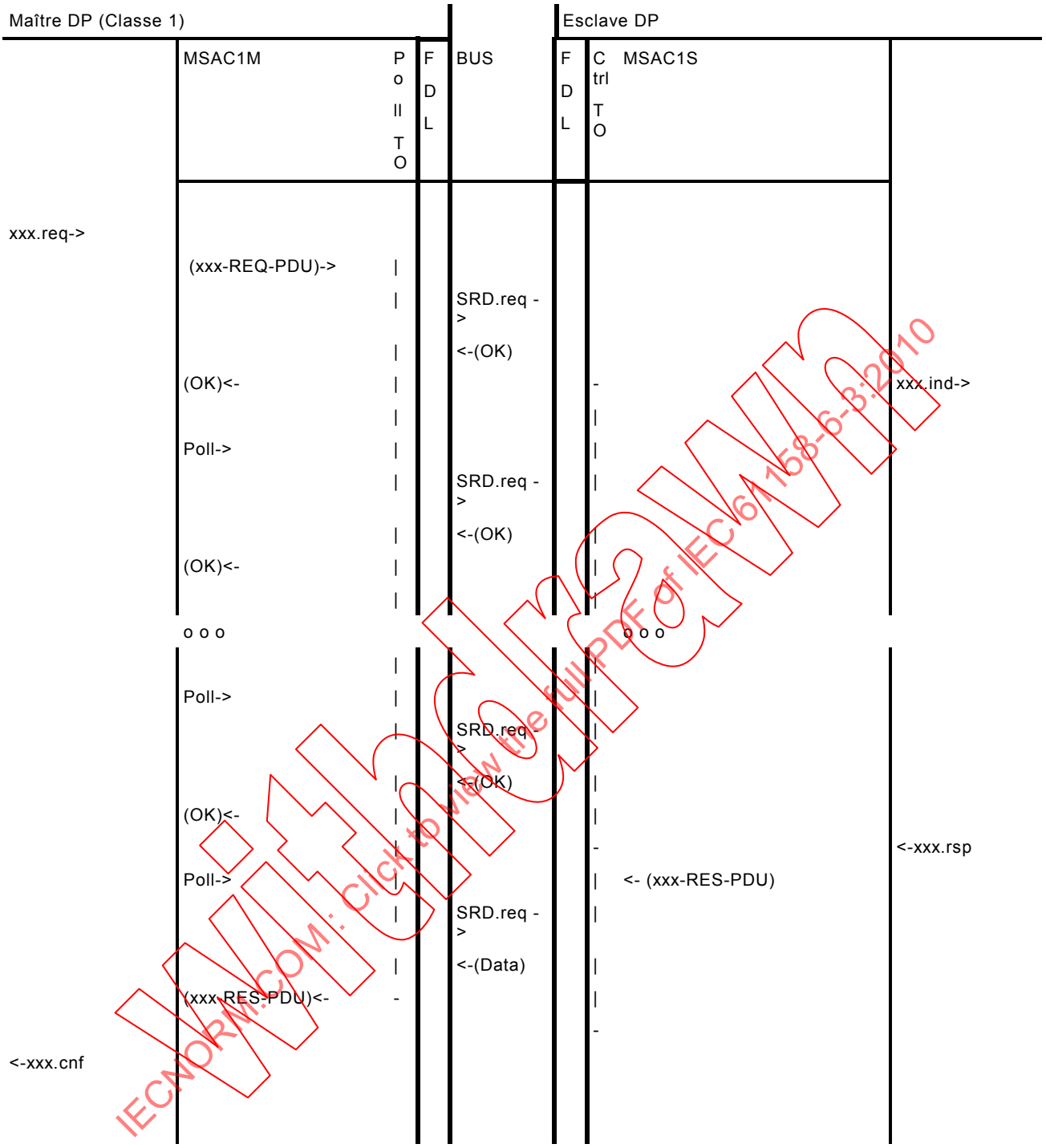
– l'apparition d'une réponse erronée,
– ou la réception d'une SRD_RES_PDU avec des données de réponse.

Le résultat sera alors transmis à l'utilisateur du maître DP (Classe 1) au moyen d'un xxx.cnf correspondant.

Le répondeur (esclave DP) attend une réponse. Si le MSAC1S de l'esclave DP reçoit une demande valide, cette demande est transmise à l'utilisateur par le xxx.ind correspondant.

Après réception de la réponse provenant de l'utilisateur (xxx.rsp), les données de réponse seront transmises avec un REPLY_UPDATE.req au DMPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn




IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Figure 23 – Séquence de communication acyclique entre maître DP (classe 1) et esclave DP**

## 6.9 Surveillance d'une relation d'applications

### 6.9.1 Surveillance de MS0 – AR

#### 6.9.1.1 Généralités

Dans les systèmes de commande industriels, il y a un besoin de vérifier le fonctionnement correct de chaque pièce individuelle. Dans un tel système, une défaillance de support ou un défaut dans un maître de bus ne doit pas entraîner des erreurs dans les dispositifs périphériques. Dans les cas susmentionnés, un dispositif périphérique doit commuter à un état de sécurité. En outre, l'utilisateur d'un maître de bus (Classe 1) doit être informé des pannes d'émission après une certaine durée.

#### 6.9.1.2 Intervalle de commande au niveau de l'esclave DP

**Commande de chien de garde**

Ce temporisateur, redémarré par des demandes reçues côté maître de bus, mettra les sorties d'un esclave de bus à l'état de sécurité après expiration du temporisateur.

**Règles pour émettre Check User Prm Result.req et Check Cfg Result.req**

L'ordre d'émission de ces primitives doit être l'ordre d'invocation des primitives Check User Prm.ind et Check Cfg Result.ind par le MSCY1S.

#### 6.9.1.3 Intervalles de commande au niveau du maître DP (Classe 1)

**Min_Slave_Interval**

Ce temps de surveillance spécifie la plus petite durée admissible entre deux cycles de sondage d'esclave. Cela assure que la séquence des demandes de fonction issues du maître de bus peut être gérée par l'esclave de bus. Cette durée sera satisfaite par le maître de bus (Classe 1) pour chaque fonction maître-esclave, avec l'exception de la fonction Global_Control. Pour la fonction Global_Control, l'utilisateur a la responsabilité de la conformité au Min_Slave_Interval.

**Data_Control_Time**

Le Data_Control_Time définit le temps dans lequel un maître DP (Classe 1) vérifie le transfert de données vers ses esclaves DP associés. De surcroît, en la moitié de cette durée, le maître DP (Classe 1) indique son mode de fonctionnement au moyen d'un service Global Control aux esclaves DP dédiés.

Le jeu de paramètres de maître contient le Data_Control_Time.

**Règle**

$$T_{WD} > T_{TR} \quad (1)$$

$$T_{WD} > \text{Min_Slave_Interval} \quad (2)$$

$$\text{Data_Control_Time} \geq 6 \cdot T_{WD} \quad (3)$$

### 6.9.2 Surveillance de MS2 – AR

L'objectif de la surveillance de connexion est de détecter la défaillance du partenaire de communication. Des PDU Idle (de repos) sont échangées pour redéclencher le temporisateur de surveillance du partenaire de communication.

Le cycle de service Idle déclenché par un maître de bus est appelé Master-Idle. Le cycle de service Idle déclenché par l'esclave est appelé Slave-Idle.

Toutes les séquences de surveillance et de repos sont commandées par une seule base de temps “Send Timeout”.

**Send_Timeout**

Chaque relation de communications MSAC2 est surveillée par un temporisateur Timer (Send_Timeout).

Si ce Timer expire, le maître de bus (Classe 2) abandonne la relation de communications MSAC2. En outre, l'utilisateur d'un maître de bus (Classe 2) doit être informé.

Ce Timer doit être installé selon

la charge du réseau,

le nombre de connexions parallèles par maître DP

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Règle**

$$T_{Send_TO(TS)} \geq \left(\sum ATCyc\right) \times Num_of_max(TS) \quad (4)$$

où

ATCyc signifie le retard de toutes les activités d'un cycle de jeton,

Num_of_max signifie nombre de connexions acycliques maximum.

NOTE Le paramètre Send_Timeout dans le service Initiate est codé comme une valeur entière non signées, chaque unité représente 10 ms. En raison des erreurs de Timer et des arrondis de nombres entiers, le prochain nombre entier possible plus un doit être utilisé comme Send_Timeout.

Il est supposé qu'une seule interaction acyclique est exécutée dans un cycle de jeton en chaque maître DP. Si plus d'une en est exécutée, le Send_TO peut être divisé par le nombre d'interactions acycliques par cycle de jeton

La Figure 24 montre un exemple pour l'établissement d'une connexion MS2 entre un Maître (Classe 2) et un Esclave. La surveillance des connexions démarre immédiatement après cela.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

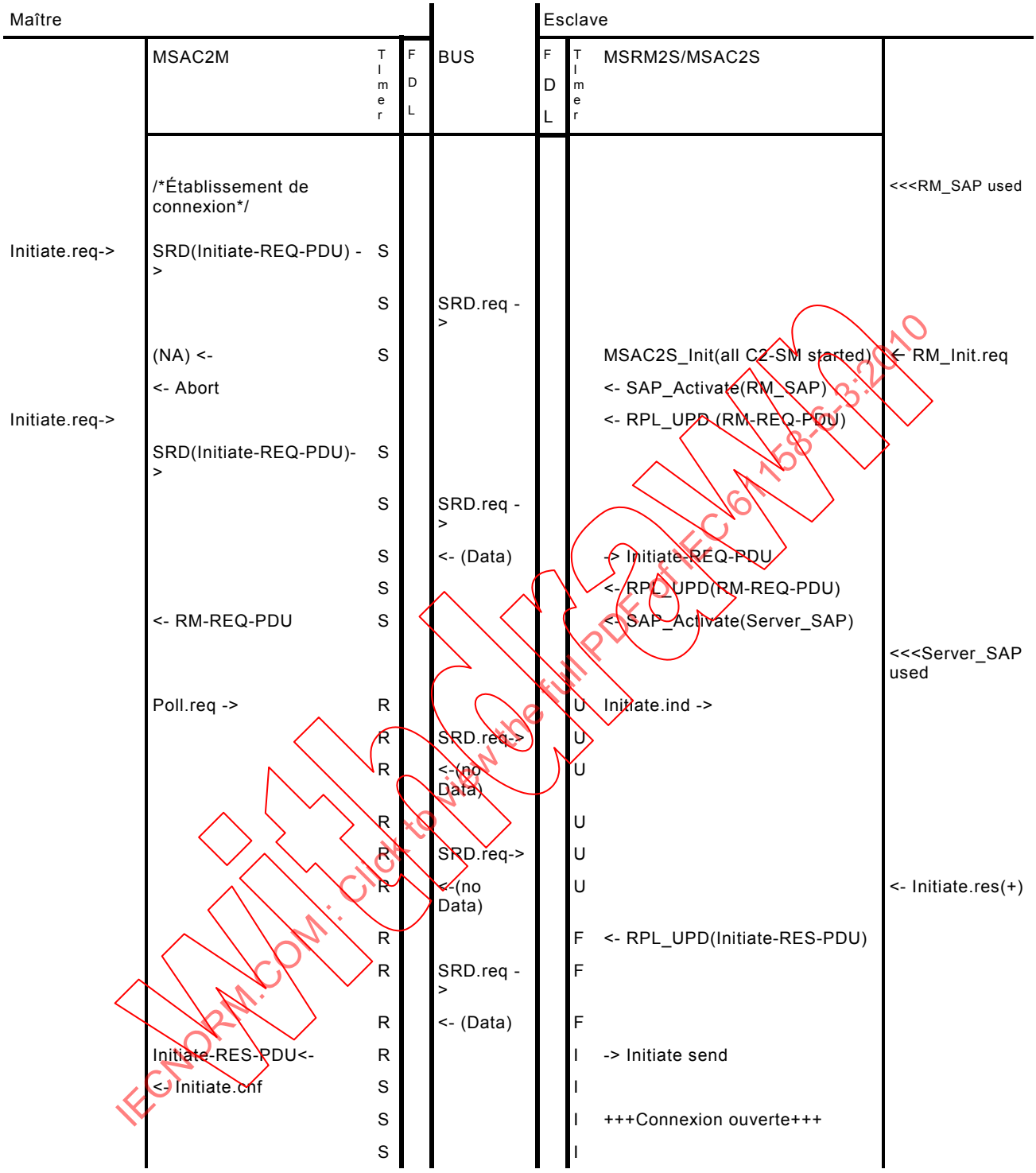



**Figure 24 – Exemple pour établissement de connexion sur MS2**

Phase a) Surveillance de connexion pendant l'attente de réponse de service MS2 en cours ou fonctionnement de Master-Idle:

Le maître DP démarre le R-Timer (temporisateur de réception «receive timer») lorsque la demande de service MS2 est transférée vers la DLL locale. Le R-Timer surveille le MSAC2S distant et la DLL locale.

Le R-Timer est arrêté si une réponse est reçue ou redémarré si une demande de Slave-Idle est reçue. La surveillance de connexion abandonne la connexion si une confirmation négative de DL apparaît ou le R-Timer expire.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

L'esclave démarre l'U-Timer (temporisateur de réponse d'utilisateur «user response timer») lorsqu'un service confirmé est transmis à l'utilisateur. L'U-Timer est arrêté lorsque l'utilisateur fournit la réponse de service. Si l'U-Timer expire une demande de Slave-Idle est envoyée pour redéclencher le R-Timer de surveillance du Maître.

L'esclave DP démarre le F-Timer (temporisateur de réponse de recherche «fetch response timer») lorsqu'une réponse de service ou une demande de Slave-Idle est transmise à la DLL locale. Le F-Timer est arrêté lorsque le maître a récupéré la PDU. Si le F-Timer expire, la surveillance de connexion abandonne la connexion.

La surveillance de connexion de phase a) est montrée à la Figure 25.

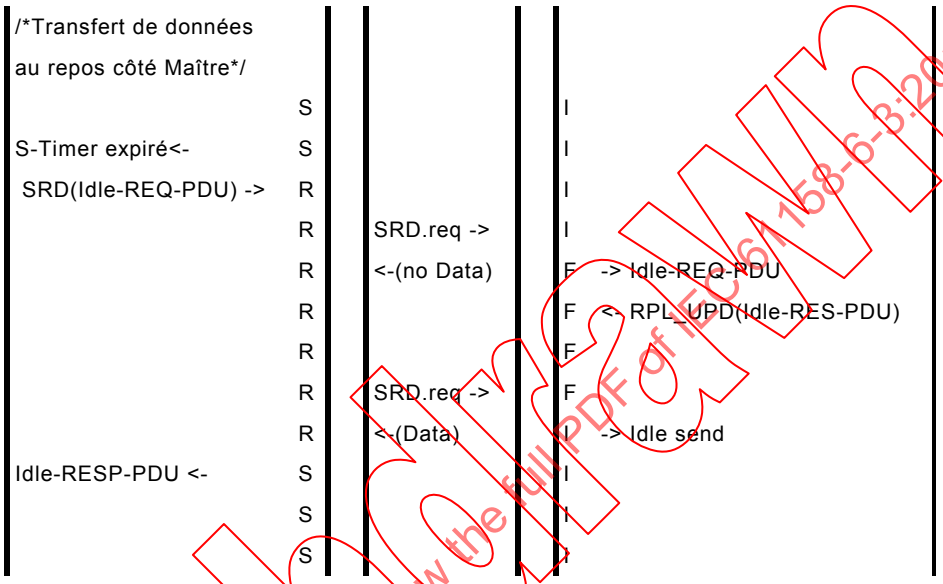



**Figure 25 – Repos côté maître sur MS2**

Phase b) Surveillance de connexion sans réponse de service en cours:

Le maître de bus démarre le S-Timer (temporisateur d'envoi «send timer») si une PDU de réponse ou une Idle-PDU est reçue. Le S-Timer est arrêté si une demande de service est fournie par l'utilisateur. Si le S-Timer expire, une demande de Master-Idle est envoyée pour arrêter l'I-Timer de surveillance de l'Esclave.

Au cours de l'attente de réponses Master-Idle en cours, la surveillance de connexion est gérée comme décrit dans la Phase a).

L'Esclave démarre l'I-Timer (temporisateur d'indication «indication timer» lorsque le tampon de mise à jour de réponses a été récupéré par le Maître. L'I-Timer est arrêté si une demande de Master-Idle ou une indication de service est reçue. Si l'I-Timer expire, la surveillance de connexion abandonne la connexion.

Le Maître surveille l'émission de l'Abort-REQ-PDU. Le Maître démarre le S-Timer lorsqu'un Abort.req est transféré à la DLL locale. Le S-Timer est arrêté lorsque l'Abort.req est envoyé. Si le S-Timer expire, la connexion est fermée et l'utilisateur est informé par un MSAC2M_Closed.ind.

L'Esclave surveille la recherche de l'Abort-REQ-PDU par le Maître. L'Esclave démarre le F-Timer lorsqu'un Abort.req est transféré à la DL locale. Le F-Timer est arrêté si le Maître a récupéré l'Abort-REQ-PDU. Si le F-Timer expire, le Server_SAP est désactivé et la connexion est fermée.

La surveillance de connexion de phase b) est montrée à la Figure 26.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Maître | | | | | Esclave | | | |
|---|---|---|---|---|---|---|---|---|
| | MSAC2M | Timer | FDL | BUS | FDL | Timer | MSRM2S/MSAC2S | |
| | /*Transfert de données | | | | | | | |
| | au repos côté Esclave*/ | | | | | | | |
| Read.req-> | | S | | | | I | | |
| | SRD(Read-REQ-PDU)-> | R | | | | I | | |
| | | R | | SRD.req -> | | I | | |
| | | R | | <-(no Data) | | | -> Read-REQ-PDU | |
| | Poll.req -> | R | | | | U | Read.ind -> | |
| | | R | | SRD.req -> | | U | | |
| | | R | | <-(no Data) | | U | | |
| | | R | | | | U | -> U-Timer expired | |
| | | R | | | | F | <- RPL_UPD(Idle-REQ-PDU) | |
| | | R | | | | F | | |
| | | R | | SRD.req -> | | F | | |
| | | R | | <-(Data) | | I | -> Idle send | |
| | Idle-REQ-PDU<- | R | | | | I | | |
| | SRD(Idle-RES-PDU) -> | R | | | | I | | |
| | | R | | SRD.req -> | | I | | |
| | | R | | <-(no Data) | | U | -> Idle-RES-PDU | |
| | | R | | | | U | | |
| | | R | | SRD.req -> | | U | | <- Read.res |
| | | R | | <-(no Data) | | F | <- RPL_UPD(Read-RES-PDU) | |
| | | R | | | | F | | |
| | | R | | SRD.req -> | | F | | |
| | | R | | <-(Data) | | I | -> Response send | |
| | Read-RES-PDU <- | R | | | | I | | |
| | <- Read.cnf | S | | | | I | | |
| | | S | | | | I | | |

**Figure 26 – Repos côté esclave sur MS2**

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

# 7 Diagramme d'états AP-Context (contexte d'AP)

Aucun diagramme d'états AP-Context n'est défini pour ce protocole.

# 8 Diagrammes protocolaires de services de la FAL (FSPM)

## 8.1 FSPMS

### 8.1.1 Définitions des primitives

#### 8.1.1.1 Primitives échangées entre FSPMS et AP-Context

Le Tableau 33 montre les primitives de service, y compris leurs paramètres associés, émises par l'AP-Context et reçues par le FSPMS.

**Tableau 33 – Primitives émises par l'AP-Context vers le FSPMS**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| DLL Init DP slave.req | AP-Context | Bus Para | Se référer à Définition des services de la couche application, Bus de terrain de Type 3, CEI 61158-5-3 |
| SInit MS0.req | AP-Context | AREP | |
| SInit MS1.req | AP-Context | AREP | |
| SInit MS2.req | AP-Context | AREP | |
| Reset DP slave.req | AP-Context | (aucun) | |
| Abort.req | AP-Context | AREP,<br>Subnet,<br>Instance,<br>Reason Code | |
| DP slave Application Ready.req | AP-Context | AREP | |
| Check User Prm Result.req | AP-Context | AREP,<br>Prm_OK | |
| Check Ext User Prm Result | AP-Context | AREP,<br>Ext_Prm_OK | |
| Check Cfg Result.req | AP-Context | AREP,<br>Cfg_OK,<br>Input Data Len,<br>Output Data Len | |
| Set Cfg.req | AP-Context | AREP,<br>Cfg Data | |
| Set Slave Diag.req | AP-Context | AREP,<br>Ext Diag Overflow,<br>Ext Diag Flag,<br>Ext Diag Data | |
| Set Input.req | AP-Context | AREP,<br>Input Data | |
| Get Output.req | AP-Context | AREP | |
| Alarm Notification.req | AP-Context | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr,<br>Add Ack,<br>Alarm Specifier,<br>Alarm Data | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Initiate.rsp(+) | AP-Context | AREP,<br>Max Len Data Unit,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Initiate.rsp(-) | AP-Context | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Read.rsp(+) | AP-Context | AREP,<br>Length,<br>Data | |
| Read.rsp(-) | AP-Context | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.rsp(+) | AP-Context | AREP,<br>Length | |
| Write.rsp(-) | AP-Context | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Data Transport.rsp(+) | AP-Context | AREP,<br>Length,<br>Data | |
| Data Transport.rsp(-) | AP-Context | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Alarm Ack.rsp(+) | AP-Context | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Alarm Ack.rsp(-) | AP-Context | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Load ARL DP slave.req | AP-Context | Min Send Timeout,<br>List of Resource Manager Entries,<br>List of S_SAP_indices,<br>List of ARL Entries | |
| Get ARL DP slave.req | AP-Context | (aucun) | |
| Load CRL DP slave.req | AP-Context | List of CRL Entries | |
| Get CRL DP slave.req | AP-Context | (aucun) | |
| Set ARL Isochronous Mode.req | AP-Context | Isochronous Mode | |
| Get Publisher Data.req | AP-Context | AREP,<br>CREP | |
| Start Subscriber.req | AP-Context | AREP,<br>CREP | |
| Stop Subsciber.req | AP-Context | AREP,<br>CREP | |
| Load CRL DXB-Linktable Entries.req | AP-Context | AREP,<br>DXB-Linktable | |
| Initiate Load.rsp(+) | AP-Context | AREP,<br>Actual LR Size,<br>Max Response Delay,<br>Max Segment Length<br>User Specific | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Initiate Load.rsp(-) | AP-Context | AREP,<br>Error Code | |
| Pull Segment.rsp(+) | AP-Context | AREP,<br>Segment Length,<br>Segment Number,<br>More Follows,<br>Data | |
| Pull Segment.rsp(-) | AP-Context | AREP,<br>Error Code | |
| Push Segment.rsp(+) | AP-Context | AREP | |
| Push Segment.rsp(-) | AP-Context | AREP,<br>Error Code | |
| Terminate Load.rsp(+) | AP-Context | AREP | |
| Terminate Load.rsp(-) | AP-Context | AREP,<br>Error Code | |
| Start.rsp(+) | AP-Context | AREP | |
| Start.rsp(-) | AP-Context | AREP<br>Error Code | |
| Stop.rsp(+) | AP-Context | AREP | |
| Stop.rsp(-) | AP-Context | AREP<br>Error Code | |
| Resume.rsp(+) | AP-Context | AREP | |
| Resume.rsp(-) | AP-Context | AREP<br>Error Code | |
| Reset.rsp(+) | AP-Context | AREP | |
| Reset.rsp(-) | AP-Context | AREP<br>Error Code | |
| Call.rsp(+) | AP-Context | AREP<br>Result Argument | |
| Call.rsp(-) | AP-Context | AREP<br>Error Code | |
| Get FI State.rsp(+) | AP-Context | AREP<br>FI State | |
| Get FI State.rsp(-) | AP-Context | AREP<br>Error Code | |

Le Tableau 34 montre les primitives de service, y compris leurs paramètres associés, émises par le FSPMS et reçues par l'AP-Context.

**Tableau 34 – Primitives émises par le FSPMS vers l'AP-Context**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Sinit DLL.cnf | FSPMS | (aucun) | Se référer à Définition des services de la couche application, Bus de terrain de Type 3, CEI 61158-5-3 |
| Sinit MS0.cnf | FSPMS | AREP | |
| Sinit MS1.cnf | FSPMS | AREP | |
| Sinit MS2.cnf | FSPMS | AREP | |
| Reset.cnf | FSPMS | (aucun) | |
| Check User Prm Result.cnf(+) | FSPMS | AREP | |
| Check User Prm Result.cnf(-) | FSPMS | AREP,<br>Status | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Check Ext User Prm Result.cnf(+) | FSPMS | AREP | |
| Check Ext User Prm Result.cnf(-) | FSPMS | AREP,<br>Status | |
| Check Cfg Result.cnf(+) | FSPMS | AREP | |
| Check Cfg Result.cnf(-) | FSPMS | AREP,<br>Status | |
| Set Cfg.cnf(+) | FSPMS | AREP | |
| Set Cfg.cnf(-) | FSPMS | AREP | |
| Set Slave Diag.cnf | FSPMS | AREP | |
| Set Input.cnf(+) | FSPMS | AREP | |
| Set Input.cnf(-) | FSPMS | AREP | |
| Get Output.cnf(+) | FSPMS | AREP,<br>Output Data,<br>Clear Flag,<br>New Flag | |
| Get Output.cnf(-) | FSPMS | AREP | |
| Started.ind | FSPMS | AREP,<br>Actual Enabled Alarms,<br>Alarm Sequence,<br>Alarm Limit | |
| Stopped.ind | FSPMS | AREP | |
| Abort.ind | FSPMS | AREP,<br>Locally Generated,<br>Subnet,<br>Instance,<br>Reason Code,<br>Additional Detail | |
| Fault.ind | FSPMS | AREP | |
| Alarm Reject.ind | FSPMS | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Set Slave Add.ind | FSPMS | AREP,<br>New Slave Add,<br>Ident Number,<br>No Add Chg,<br>Rem Slave Data | |
| Check User Prm.ind | FSPMS | AREP,<br>User Prm Data | |
| Check Ext User Prm.ind | FSPMS | AREP,<br>Ext User Prm Data | |
| Check Cfg.ind | FSPMS | AREP,<br>Cfg Data | |
| New Output.ind | FSPMS | AREP,<br>Output Data,<br>Clear Flag | |
| Global Control.ind | FSPMS | AREP,<br>Clear Command,<br>Sync Command,<br>Freeze Command,<br>Group Select | |
| Initiate.ind | FSPMS | AREP,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Read.ind | FSPMS | AREP,<br>Slot Number,<br>Index,<br>Length | |
| Write.ind | FSPMS | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Data Transport.ind | FSPMS | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Alarm Ack.ind | FSPMS | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Load ARL DP slave.cnf(+) | FSPMS | (aucun) | |
| Load ARL DP slave.cnf(-) | FSPMS | Status | |
| Get ARL DP slave.cnf(+) | FSPMS | Min Send Timeout,<br>List of Resource Manager Entries,<br>List of S_SAP_indices,<br>List of ARL Entries | |
| Get ARL DP slave.cnf(-) | FSPMS | Status | |
| Load CRL DP slave.cnf(+) | FSPMS | (aucun) | |
| Load CRL DP slave.cnf(-) | FSPMS | Status | |
| Get CRL DP slave.cnf(+) | FSPMS | List of CRL Entries | |
| Get CRL DP slave.cnf(-) | FSPMS | Status | |
| Set ARL Isochronous Mode.cnf(+) | FSPMS | | |
| Set ARL Isochronous Mode.cnf(-) | FSPMS | Status | |
| SYNCH Event.ind | FSPMS | AREP,<br>Status | |
| Publisher Active.ind | FSPMS | AREP,<br>CREP,<br>Status | |
| New Publisher Data.ind | FSPMS | AREP,<br>CREP | |
| Get Publisher Data.cnf(+) | FSPMS | AREP,<br>CREP,<br>Data,<br>New Flag | |
| Get Publisher Data.cnf(-) | FSPMS | AREP,<br>CREP | |
| Start Subsciber.cnf(+) | FSPMS | AREP,<br>CREP | |
| Start Subscriber.cnf(-) | FSPMS | AREP,<br>CREP | |
| Stop Subscriber.cnf(+) | FSPMS | AREP,<br>CREP | |
| Stop Subscriber.cnf(-) | FSPMS | AREP,<br>CREP | |
| Load CRL DXB-Linktable Entries.cnf(+) | FSPMS | AREP | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Load CRL DXB-Linktable Entries.cnf(-) | FSPMS | AREP | |
| Set Time.ind | FSPMS | AREP,<br>Time Value,<br>Local Time Diff,<br>Summertime,<br>Accuracy,<br>Synchronisation Active,<br>Announcement Hour | |
| Sync Interval Violation.ind | FSPMS | AREP | |
| Initiate Load.ind | FSPMS | AREP,<br>Slot Number,<br>LR Index,<br>Load Type,<br>Load Image Size,<br>Intersegment Request Timeout<br>User Specific | |
| Pull Segment.ind | FSPMS | AREP,<br>Slot Number,<br>LR Index,<br>Segment Length | |
| Push Segment.ind | FSPMS | AREP,<br>Slot Number<br>LR Index,<br>Segment Length<br>Segment Number,<br>More Follows,<br>Data | |
| Terminate Load.ind | FSPMS | AREP<br>Slot Number<br>LR Index | |
| Start.ind | FSPMS | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Stop.ind | FSPMS | AREP<br>Slot Number<br>FI Index | |
| Resume.ind | FSPMS | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Reset.ind | FSPMS | AREP<br>Slot Number<br>FI Index | |
| Call.ind | FSPMS | AREP<br>Slot Number<br>Entity Number<br>FI Index<br>Execution Argument | |
| Get FI State | FSPMS | AREP<br>Slot Number<br>FI Index | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

#### 8.1.1.2 Paramètres des primitives de FSPMS

Les paramètres utilisés avec les primitives échangées entre le FSPMS et l'AP-Context sont décrits dans "Définition des services de la FAL, Bus de terrain de Type 3" (voir la CEI 61158-5-3).

### 8.1.2 Description de diagramme d'états

Ce Diagramme est utilisé pour coordonner les interactions entre l'AP-Context (Contexte d'AP) et les DMPMS, MSCY1S, MSAC1S, MSRM2S et MSAC2S. Comme les alarmes exigent la prise en charge de plusieurs diagrammes d'états, les alarmes sont gérées par ce diagramme.

Avec le SInit_MS1.req, les variables locales des diagrammes d'états sont mises aux valeurs de démarrage et les diagrammes d'états MSAC1 sont initialisés.

Lorsque le MSCY1S entre dans l'état d'échange de données "Data-Exchange-State" (ce qui est signalé par un start.ind), des messages d'alarme peuvent être générés par l'utilisateur.

Le traitement parallèle des messages d'alarme est limité par:

le nombre des classes d'alarme "Alarm_Class" (Alarm_Sequence = False)

le nombre des Alarmes acceptées par le Maître (Alarm_Sequence = True)

le nombre des Alarmes gérées par cet Esclave (Alarm_Sequence = True).

Les alarmes peuvent être traitées uniquement si l'Alarm_Class appropriée est activée par le maître DP. Une alarme produite par une Alarm_Notification sera émise par le biais d'un diagnostic (MSCY1S) vers un maître DP (Classe 1). L'acquittement sera reçu par un service Alarm Ack traité par le MSAC1S.

Le diagramme d'états a besoin des variables locales qui sont décrites ci-dessous. En outre, le diagramme utilise des fonctions et des macros qui sont énumérées en 8.1.4.

**Variables locales**

**Alarm_Sequence**
**(Boolean)**
Cette variable indique si,

une seule alarme d'une Alarm_Class spécifique peut être active à un instant donné (Alarm_Sequence = False), ou

plusieurs alarmes (2à 32) de n'importe quel type peuvent être actives à un instant donné (Alarm_Sequence = True).

**Initial_Alarm_Sequence**
(Boolean)

Cette variable stocke la capacité de base de l'Esclave.

**Outstanding_Alarm_TABLE**
(Array 0…7, 0…31 of Alarm_Status, Alarm_Type)

Alarm_Status: pending, not_pending

Alarm_Type: 1 à 6, 32 à 126

L'Alarm_State_Table est une matrice bidimensionnelle ayant 7 * 32 éléments. Chaque élément de la matrice stocke des informations relatives à l'état actuel de n'importe quelle alarme. La matrice est adressée avec l'Alarm_Class calculée à partir de l'Alarm_Type et du Seq_Nr de l'alarme.

**Alarm_Max**
(Unsigned8)

Cette variable indique le nombre maximum d'alarmes parallèles de cet Esclave.

Plage: 1..32

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Alarm_Limit**
(Unsigned8)

Cette variable contient le nombre maximum d'alarmes autorisées par la connexion Maître-Esclave réelle.

Plage: 1..32

**Alarm_Decode**
(Array 0…7 of Unsigned8)

Décodage du nombre d'alarmes parallèles:

Alarm_Decode[0] = 1

Alarm_Decode[1] = 2

Alarm_Decode[2] = 4

Alarm_Decode[3] = 8

Alarm_Decode[4] = 12

Alarm_Decode[5] = 16

Alarm_Decode[6] = 24

Alarm_Decode[7] = 32

**Alarm_Count**
(Unsigned8)

Ce compteur contient le nombre d'alarmes (0..32) qui ont été réellement envoyées par l'esclave DP. Le compteur est uniquement utilisé dans le mode de séquence.

Plage: 0 à Alarm_Limit

**Req_Alarm_FIFO**
Le Req_Alarm_FIFO est utilisé pour stocker les alarmes qui doivent encore être envoyées. Chaque élément consiste en une Alarm-PDU. Le FIFO doit être capable de contenir jusqu'à 32 respectivement Alarm_Max entrées.

Actual_Enabled_Alarms

(Array of Bits)

Cette variable indique les types d'alarmes qui sont effectivement pris en charge par le Maître.

Bit 0 = réservé

Bit 1 = réservé

Bit 2 = Update_Alarm

Bit 3 = Status_Alarm

Bit 4 = Manufacturer_Specific_Alarm

Bit 5 = Diagnostic_Alarm

Bit 6 = Process_Alarm

Bit 7 = Pull_Plug_Alarm

**Alarms_Supported**
(Array of Bits)

Cette variable indique les types d'alarmes qui sont pris en charge par l'Esclave.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Bit 0 = réservé

Bit 1 = réservé

Bit 2 = Update_Alarm

Bit 3 = Status_Alarm

Bit 4 = Manufacturer_Specific_Alarm

Bit 5 = Diagnostic_Alarm

Bit 6 = Process_Alarm

Bit 7 = Pull_Plug_Alarm

**Local_Diag_Buffer**
(Octet-String)

Cette variable locale est utilisée pour stocker les Diag-Data (données de dialogue), à l'exception des 6 premiers octets et les PDU d'alarme et de statut.

**L_Ext_Diag_Flag**
(Bit)

Cette variable locale est utilisée pour stocker l'Ext_Diag_Flag.

**Stored_Diag**
(Boolean)

Fanion qui indique que de nouvelles Diag_Data sont stockées dans le tampon Diag_Buffer pendant l'attente de l'émission d'un précédent diagnostic .

**Diag_Buffer**

(Octet-String)

Tampon pour stocker un Slave-Diagnostic(Identification-Related-Diagnosis, Channel-Related-Diagnosis, Revision_Number)

**Index**
(Structure of Alarm_Class, Seq_Nr)

Cet indice est utilisé pour adresser l'Alarm_Table en cours.

**LR_State**
(Unsigned8)

Cette variable stocke l'état de la séquence Load Region.

**CurrentBuffer**
(Array of structure Empty, WriteLength and Data)

Ce tampon stocke des APDU pour Load Region pour chaque AREP.

**CurrentLoadType**
(Unsigned8)

Cette variable stocke si une séquence Pull ou Push est active.

**CurrentLRIndex**
(Unsigned16)

Cette variable stocke le type LR.

### 8.1.3 Table d'états de FSPMS

Le Tableau 35 contient la description complète du diagramme d'états FSPMPS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Contraintes de traitement pour le mode isochrone:**
Le temps entre l'émission du SYNCH.ind jusqu'aux demandes de service du MSCY1S à la DL doit être suffisamment court pour permettre à la MAC d'envoyer toutes les demandes issues de MSCY1S en un cycle.

La définition suivante pour un ensemble d'états est utilisée dans le Tableau 35.

t_state = W-START ou W-DIA-UPD ou W-FETCHED

**Tableau 35 – Table d'états de FSPMS**

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 1 | POWER-ON | Load ARL DP Slave.req (Min Send Timeout, List of Resource Manager Entries, List of S_SAP_indices, List of ARL Entries)<br>/paramètres valides<br>=><br>Load ARL DP Slave.cnf(+) | POWER-ON |
| 2 | POWER-ON | Load ARL DP Slave.req (Min Send Timeout, List of Resource Manager Entries, List of S_SAP_indices, List of ARL Entries)<br>/paramètres non valides<br>=><br>Status:= NO<br>Load ARL DP Slave.cnf(-) (Status) | POWER-ON |
| 3 | POWER-ON | Get ARL DP Slave.req<br>/ARL loaded<br>=><br>Get ARL DP Slave.cnf(+) ( Min Send Timeout, List of Resource Manager Entries, List of S_SAP_indices, List of ARL Entries ) | POWER-ON |
| 4 | POWER-ON | Get ARL DP Slave.req<br>/ARL not loaded<br>=><br>Status:= NO<br>Get ARL DP Slave.cnf(-) (Status) | POWER-ON |
| 5 | POWER-ON | Load CRL DP Slave.req ( List of CRL Entries )<br>/paramètres non valides<br>=><br>Load CRL DP Slave.cnf(+) | POWER-ON |
| 6 | POWER-ON | Load CRL DP Slave.req ( List of CRL Entries )<br>/paramètres non valides<br>=><br>Status:= NO<br>Load CRL DP Slave.cnf(-) (Status) | POWER-ON |
| 7 | POWER-ON | Get CRL DP Slave.req<br>/CRL loaded<br>=><br>Get CRL DP Slave.cnf(+) (List of CRL Entries) | POWER-ON |
| 8 | POWER-ON | Get CRL DP Slave.req<br>/CRL not loaded<br>=><br>Status:= NO<br>Get CRL DP Slave.cnf(-) (Status) | POWER-ON |
| 9 | POWER-ON | FSPMS_SInit_MS1.req (AREP)<br>/Alarm_Mode_Slave <> 0<br>=><br>Alarm_Sequence:= True<br>Initial_Alarm_Sequence:= Alarm_Sequence<br>Alarm_Max:= Alarm_Decode[Alarm_Mode_Slave]<br>Alarm_Max:= max(Alarm_Max, 7)<br>Local_Diag_Buffer:= Default_Ext_Diag_Data<br>L_Ext_Diag_Flag:= Default_Ext_Diag<br>FILL (Outstanding_Alarm_TABLE.Alarm_Status, not_pending)<br>Stored_Diag:=False<br>Reset_Req_Alarm_FIFO()<br>Act_Ref:= 0<br>MSAC1_SInit_MS1.req | W-START-C1 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 10 | POWER-ON | FSPMS_SInit_MS1.req (AREP)<br>/Alarm_Mode_Slave = 0<br>=><br>Alarm_Sequence:= False<br>Initial_Alarm_Sequence:= Alarm_Sequence<br>Alarm_Max:= 7<br>Local_Diag_Buffer:= Default_Ext_Diag_Data<br>L_Ext_Diag_Flag:= Default_Ext_Diag<br>FILL(Outstanding_Alarm_TABLE.Alarm_Status, not_pending)<br>Stored_Diag:=False<br>Reset_Req_Alarm_FIFO()<br>FSPMS_User_Reset:=False<br>Act_Ref:= 0<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSAC1_SInit_MS1.req | W-START-C1 |
| 11 | W-START-C1 | MSAC1S_SInit_MS1.cnf<br>=><br>FSPMS_SInit_MS1.cnf(AREP) | W-START |
| 12 | W-START | FSPMS_Abort.req(AREP)<br>/AREP.AR_Type=MS0<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Abort.req | W-START |
| 13 | W-START | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP, Alarm_Type, Slot_Number, Seq_Nr, Status=No_Start) | W-START |
| 14 | W-START | FSPMS_Set_Slave_Diag.req(AREP,Ext_Diag_Flag, Ext_Diag)<br>=><br>L_Ext_Diag_Flag:= Ext_Diag_Flag<br>L_Ext_Diag_Overflow:= Ext_Diag_Overflow<br>Local_Diag_Buffer:= Ext_Diag<br>Act_Ref:= Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref)<br>FSPMS_Set_Slave_Diag.cnf(AREP) | W-START |
| 15 | W-START | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>=><br>MSCY1S_Abort.req<br>FSPMS_Abort.ind(AREP) | W-START |
| 16 | W-START | MSCY1S_Set_Slave_Diag.cnf(-) (Reference)<br>=><br>ignore | W-START |
| 17 | W-START | MSCY1S_Set_Slave_Diag.cnf(+) (Reference)<br>=><br>ignore | W-START |
| 18 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/(Enabled Alarms & Alarms_Supported) = 0<br>=><br>FSPMS DP Slave Started.ind(AREP,Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | W-START |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 19 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/ ((Enabled Alarms & Alarms_Supported) <> 0) && Alarm_Mode_Master <> 0 && Initial_Alarm_Sequence = TRUE && Diag_Flag=FALSE<br>=><br>Alarm_Sequence:= True<br>Actual_Enabled_Alarms:= Enabled_Alarms<br>Alarm_Limit:= min(Alarm_Max, Alarm_Decode[Alarm_Mode_Master])<br>Alarm_Count:= 0<br>FSPMS DP Slave Started.ind(AREP,Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | W-DIA-UPD |
| 20 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/ ((Enabled Alarms & Alarms_Supported) <> 0) && Alarm_Mode_Master <> 0 && Initial_Alarm_Sequence =FALSE<br>=><br>MSCY1S_Abort.req | W-START |
| 21 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/ ((Enabled Alarms & Alarms_Supported) <> 0) && Alarm_Mode_Master = 0 && Diag_Flag=TRUE<br>=><br>Alarm_Sequence:= False<br>Actual_Enabled_Alarms:= Enabled_Alarms<br>FSPMS DP Slave Started.ind(AREP,Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | W-FETCHED |
| 22 | W-START | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>/ ((Enabled Alarms & Alarms_Supported) <> 0) && Alarm_Mode_Master = 0 && Diag_Flag=FALSE<br>=><br>Alarm_Sequence:= False<br>Actual_Enabled_Alarms:= Enabled_Alarms<br>FSPMS DP Slave Started.ind(AREP,Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | W-DIA-UPD |
| 23 | W-START | MSCY1S_Stop.ind<br>=><br>FSPMS_Stopped.ind(AREP) | W-START |
| 24 | W-DIA-UPD | FSPMS_Abort.req(AREP)<br>/AREP.AR_Type=MS0<br>=><br>Index:= Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Abort.req | RESET-P-LEAVE |
| 25 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=FALSE<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Not_Enabled) | W-DIA-UPD |
| 26 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE && Alarm_Count=Alarm_Limit<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP, Alarm_Type, Slot_Number, Seq_Nr, Status=Limit_Expired) | W-DIA-UPD |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 27 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE &&<br>Alarm_Count<Alarm_Limit<br>&&<br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]=not_pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=pending<br>Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]:=Alarm_Type<br>Alarm_Count:=Alarm_Count+1<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer, Alarm(Slot_Number,Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data))<br>Act_Ref:= Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref)<br>FSPMS_Alarm_Notification.cnf(+)(AREP,Alarm_Type, Slot_Number, Seq_Nr) | W-FETCHED |
| 28 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE &&<br>Alarm_Count<Alarm_Limit<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= pending<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Sequence_Nr_Pending) | W-DIA-UPD |
| 29 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= pending<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Sequence_Nr_Pending) | W-DIA-UPD |
| 30 | W-DIA-UPD | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= not_pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=pending<br>Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]:=Alarm_Type<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer, Alarm(Slot_Number,Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data))<br>Act_Ref:= Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref)<br>FSPMS_Alarm_Notification.cnf(+)(AREP,Alarm_Type, Slot_Number, Seq_Nr) | W-FETCHED |
| 31 | W-DIA-UPD | FSPMS_Set_Slave_Diag.req(AREP,Ext_Diag_Flag, Ext_Diag)<br>=><br>if ( PrmCmdAck in Ext_Diag or Stored_PrmCmdAck != NULL )<br>Stored_PrmCmdAck:= PrmCmdAck, Ext_Diag.Red_Status:= PrmCmdAck endif<br>L_Ext_Diag_Flag:= Ext_Diag_Flag<br>L_Ext_Diag_Overflow:= Ext_Diag_Overflow<br>Local_Diag_Buffer:= Ext_Diag<br>Act_Ref:= Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref)<br>FSPMS_Set_Slave_Diag.cnf(AREP) | W-DIA-UPD |
| 32 | W-DIA-UPD | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Alarm_Sequence=TRUE &&<br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]=pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:= not_pending<br>Alarm_Count:=Alarm_Count-1<br>FSPMS_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>MSAC1S_Alarm Ack.rsp(+) ( Slot Number, Alarm Type, Seq Nr ) | W-DIA-UPD |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 33 | W-DIA-UPD | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Alarm_Sequence=FALSE &&<br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr] =pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:= not_pending<br>FSPMS_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>MSAC1S_Alarm Ack.rsp(+) ( Slot Number, Alarm Type, Seq Nr ) | W-DIA-UPD |
| 34 | W-DIA-UPD | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr] =not_pending<br>=><br>Index:= Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>MSCY1S_Abort.req<br>FSPMS_Abort.ind(AREP) | RESET-P-LEAVE |
| 35 | W-DIA-UPD | MSCY1S_Set_Slave_Diag.cnf(-) (Reference)<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 36 | W-DIA-UPD | MSCY1S_Set_Slave_Diag.cnf(+) (Reference)<br>/Reference < Act_Ref<br>=><br>ignore | W-DIA-UPD |
| 37 | — | — | — |
| 38 | W-DIA-UPD | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>=><br>MSCY1S_Abort.req | W-START |
| 39 | W-DIA-UPD | MSCY1S_Stop.ind<br>=><br>Alarm_Count:=0<br>Index:= Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending) | RESET-PENDING |
| 40 | RESET-PENDING | /Index.Seq_Nr <> 255<br>=><br>Outstanding Alarm_TABLE.Alarm_Status[Index]:= not_pending<br>Alarm_Type:= Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]<br>Index:= Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number=0,<br>Index.Seq_Nr, Status=MSAL1S_Stopped) | RESET-PENDING |
| 41 | RESET-PENDING | /Index.Seq_Nr = 255<br>=><br>Reset_Req_Alarm_FIFO()<br>FSPMS DP Slave Stopped.ind(AREP) | W-START |
| 42 | RESET-P-LEAVE | /Index.Seq_Nr <> 255<br>=><br>Outstanding Alarm_TABLE.Alarm_Status[Index]:= not_pending<br>Alarm_Type:= Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]<br>Index:= Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number=0,<br>Index.Seq_Nr, Status=Stopped) | RESET-P-LEAVE |
| 43 | RESET-P-LEAVE | /Index.Seq_Nr = 255<br>=><br>Reset_Req_Alarm_FIFO() | W-START |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 44 | W-FETCHED | FSPMS_Abort.req(AREP)<br>/AREP.AR_Type=MS0 && Alarm_Sequence=TRUE && Alarm_Count=0<br>=><br>Stored_Diag:=False<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer)Act_Ref:= Act_Ref + 1<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref)<br>MSCY1S_Abort.req | W-START |
| 45 | W-FETCHED | FSPMS_Abort.req(AREP)<br>/AREP.AR_Type=MS0 && Alarm_Sequence=FALSE \|\| Alarm_Count<>0<br>=><br>Stored_Diag:=False<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer)<br>Index:= Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>Act_Ref:= Act_Ref + 1<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref)<br>MSCY1S_Abort.req | RESET-P-LEAVE |
| 46 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=FALSE<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Not_Enabled) | W-FETCHED |
| 47 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE && Alarm_Count=Alarm_Limit<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Limit_Expired) | W-FETCHED |
| 48 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE && Alarm_Count<Alarm_Limit<br>&&<br>Outstanding_Alarm_TABLE.Alarm_Status[AclsAlarm_Type),Seq_Nr]=not_pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=pending<br>Alarm_Count:=Alarm_Count+1<br>Store_to_Req_Alarm_FIFO(Alarm(Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data))<br>FSPMS_Alarm_Notification.cnf(+)(AREP,Alarm_Type, Slot_Number, Seq_Nr) | W-FETCHED |
| 49 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=TRUE && Alarm_Count<Alarm_Limit&&<br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= pending<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP,Alarm_Type, Slot_Number, Seq_Nr, Status=Sequence_Nr_Pending) | W-FETCHED |
| 50 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]= pending<br>=><br>FSPMS_Alarm_Notification.cnf(-)(AREP, Alarm_Type, Slot_Number, Seq_Nr, Status=Sequence_Nr_Pending) | W-FETCHED |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 51 | W-FETCHED | FSPMS_Alarm_Notification.req(AREP, Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data)<br>/ALARM_ENABLED=TRUE && Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[AclsL(Alarm_Type),Seq_Nr]= not_pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[AclsL(Alarm_Type),Seq_Nr]:= pending<br>Outstanding_Alarm_TABLE.Alarm_Type[Acls(Alarm_Type),Seq_Nr]:= Alarm_Type<br>Store_to_Req_Alarm_FIFO(Alarm(Slot_Number, Alarm_Type, Seq_Nr, Add_Ack, Alarm_Specifier, Alarm_Data))<br>FSPMS_Alarm_Notification.cnf(+)(AREP,Alarm_Type, Slot_Number, Seq_Nr) | W-FETCHED |
| 52 | W-FETCHED | FSPMS_Set_Slave_Diag.req(AREP,Ext_Diag_Flag, Ext_Diag)<br>=><br>if ( PrmCmdAck in Ext_Diag or Stored_PrmCmdAck != NULL )<br>Stored_PrmCmdAck:= PrmCmdAck, Ext_Diag.Red_Status:= PrmCmdAck endif<br>Stored_Diag:= True<br>L_Ext_Diag_Flag:= Ext_Diag_Flag<br>L_Ext_Diag_Overflow:= Ext_Diag_Overflow<br>Local_Diag_Buffer:= Ext_Diag<br>FSPMS_Set_Slave_Diag.cnf(AREP) | W-FETCHED |
| 53 | W-FETCHED | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Alarm_Sequence=TRUE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls (Alarm_Type),Seq_Nr]=pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=not_pending<br>Alarm_Count:=Alarm_Count-1<br>FSPMS_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>MSAC1S_Alarm Ack.rsp(+) ( Slot Number, Alarm Type, Seq Nr ) | W-FETCHED |
| 54 | W-FETCHED | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Alarm_Sequence=FALSE<br>&& Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]=pending<br>=><br>Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]:=not_pending<br>FSPMS_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>MSAC1S_Alarm Ack.rsp(+) ( Slot Number, Alarm Type, Seq Nr ) | W-FETCHED |
| 55 | W-FETCHED | MSAC1S_Alarm_Ack.ind(Slot_Number, Alarm_Type, Seq_Nr)<br>/Outstanding_Alarm_TABLE.Alarm_Status[Acls(Alarm_Type),Seq_Nr]=not_pending<br>=><br>Index:= Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>MSCY1S_Abort.req<br>FSPMS_Abort.ind(AREP) | RESET-P-LEAVE |
| 56 | W-FETCHED | MSCY1S_Set_Slave_Diag.cnf(-) (Reference)<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 57 | W-FETCHED | MSCY1S_Set_Slave_Diag.cnf(+) (Reference)<br>/Reference < Act_Ref<br>=><br>ignore | W-FETCHED |
| 58 | W-FETCHED | MSCY1S_Set_Slave_Diag.cnf(+) (Reference)<br>/Reference >= Act_Ref<br>=><br>Stored_PrmCmdAck:= NULL | CHK-TRANSMIT |
| 59 | W-FETCHED | MSCY1S_Start.ind(Enabled_Alarms, Alarm_Mode_Master, Diag_Flag)<br>=><br>MSCY1S_Abort.req | W-START |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 60 | W-FETCHED | MSCY1S_Stop.ind<br>/Alarm_Sequence=TRUE && Alarm_Count=0<br>=><br>Stored_Diag:=False<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer)<br>Act_Ref:= Act_Ref + 1<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref)<br>FSPMS_Stopped.ind(AREP) | W-START |
| 61 | W-FETCHED | MSCY1S_Stop.ind<br>/Alarm_Sequence=FALSE \|\| Alarm_Count<>0<br>=><br>Stored_Diag:=False<br>Alarm_Count:=0<br>Index:= Search_In_Outstanding_Alarm_TABLE(Alarm_Status:=pending)<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer)<br>Act_Ref:= Act_Ref + 1<br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref) | RESET-PENDING |
| 62 | CHK-TRANSMIT | /Req_Alarm_FIFO_state=empty &&Stored_Diag = FALSE | W-DIA-UPD |
| 63 | CHK-TRANSMIT | /Req_Alarm_FIFO_state=empty && Stored_Diag = TRUE=>Stored_Diag:=FalseExt_Diag_Flag:=L_Ext_Diag_FlagExt_Diag_Overflow:= L_Ext_Diag_OverflowExt_Diag:=(Local_Diag_Buffer)Act_Ref:= Act_Ref + 1MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref) | W-FETCHED |
| 64 | CHK-TRANSMIT | /Req_Alarm_FIFO_state<>empty<br>=><br>Alarm:=Load_from_Req_Alarm_FIFO()<br>Stored_Diag:=False<br>Ext_Diag_Flag:=L_Ext_Diag_Flag<br>Ext_Diag_Overflow:=L_Ext_Diag_Overflow<br>Ext_Diag:=(Local_Diag_Buffer, Alarm)<br>Act_Ref:= Act_Ref + 1<br>MSCY1S_Set_Slave_Diag.req(Ext_Diag_Flag, Ext_Diag_Overflow, Ext_Diag, Reference:= Act_Ref) | W-FETCHED |
| 65 | W-RESET | DMPMS_Reset.cnf<br>/R_Cnt<3<br>=><br>R_Cnt:=R_Cnt+1 | W-RESET |
| 66 | W-RESET | MSCY1S_Reset.cnf<br>/R_Cnt<3<br>=><br>R_Cnt:=R_Cnt+1 | W-RESET |
| 67 | W-RESET | MSAC1S_Reset.cnf<br>/R_Cnt<3<br>=><br>R_Cnt:=R_Cnt+1 | W-RESET |
| 68 | W-RESET | MSRM2S_Reset.cnf<br>/R_Cnt<3<br>=><br>R_Cnt:=R_Cnt+1 | W-RESET |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 69 | W-RESET | DMPMS_Reset.cnf<br>/R_Cnt>=3<br>=><br>if (FSPMS_User_Reset=True)<br>FSPMS_Reset DP Slave .cnf<br>endif | POWER-ON |
| 70 | W-RESET | MSCY1S_Reset.cnf<br>/R_Cnt>=3<br>=><br>if (FSPMS_User_Reset=True)<br>FSPMS_Reset DP Slave .cnf<br>endif | POWER-ON |
| 71 | W-RESET | MSAC1S_Reset.cnf<br>/R_Cnt>=3<br>=><br>if (FSPMS_User_Reset=True)<br>FSPMS_Reset DP Slave .cnf<br>endif | POWER-ON |
| 72 | W-RESET | MSRM2S_Reset.cnf<br>/R_Cnt>=3<br>=><br>if (FSPMS_User_Reset=True) FSPMS_Reset DP Slave .cnfendif | POWER-ON |
| 73 | tout état | DMPMS_SYCL_Clock_Value.ind (Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg)<br>=><br>FSPMS_SYCL_Clock_Value.ind (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg) | SAME |
| 74 | tout état | DMPMS_Fault.ind<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 75 | tout état | MSAC1S_Fault.ind<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 76 | tout état | MSCY1S_Fault.ind<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 77 | tout état | MSRM2S_Fault.ind<br>=><br>R_Cnt:=0<br>FSPMS DP Slave Fault.ind<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 78 | tout état | FSPMS_Reset DP Slave .req<br>=><br>R_Cnt:=0<br>FSPMS_User_Reset:=True<br>DMPMS_Reset.req<br>MSCY1S_Reset.req<br>MSAC1S_Reset.req<br>MSRM2S_Reset.req | W-RESET |
| 79 | W-FETCHED | Set_ARL_Isochron_Mode.req(Isochron_Mode)<br>=><br>Set_ARL_Isochron_Mode.cnf(+) | W-FETCHED |
| 80 | W-DIA-UPD | Set_ARL_Isochron_Mode.req(Isochron_Mode)<br>=><br>Set_ARL_Isochron_Mode.cnf(+) | W-DIA-UPD |
| 81 | W-START | Set_ARL_Isochron_Mode.req(Isochron_Mode)<br>=><br>Status:= NO<br>Set_ARL_Isochron_Mode.cnf(-)(Status) | W-START |
| 82 | t-state | Load CRL DXB Linktable Entries.req(AREP, DXB-Linktable)<br>//Paramètres valides<br>=><br>Load CRL DXB Linktable Entries.cnf(+) | |
| 83 | t-state | Load CRL DXB Linktable Entries.req(AREP, DXB-Linktable)<br>/Paramètres non valides<br>=><br>Status:= NO<br>Load CRL DXB Linktable Entries.cnf(-) (Status) | |
| 84 | t-state | FSPMS DLL Init DP Slave.req ( Bus Para )<br>=><br>DMPMS_SInit_DLL.req ( Bus_Para ) | SAME |
| 85 | t-state | FSPMS_SInit MS0.req ( AREP )<br>=><br>MSCY1S_SInit MS0.req ( ) | SAME |
| 86 | t-state | FSPMS_SInit MS2.req ( )<br>=><br>MSRM2S_SInit MS2.req ( ) | SAME |
| 87 | t-state | FSPMS_Abort.req ( AREP, Subnet, Instance, Reason Code )<br>/AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>MSAC2S_Abort.req ( Subnet, Instance, Reason Code ) | SAME |
| 88 | t-state | FSPMS_DP Slave Application Ready.req ( AREP )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Application Ready.req ( ) | SAME |
| 89 | t-state | FSPMS_Check User Prm Result.req ( AREP, Prm_OK )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Check User Prm Result.req ( Prm_OK ) | SAME |
| 90 | t-state | FSPMS_Check Cfg Result.req ( AREP, Cfg_OK, Input Data Len, Output Data Len )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len ) | SAME |
| 91 | t-state | FSPMS_Set Cfg.req ( AREP, Cfg Data )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Set Cfg.req ( Cfg Data ) | SAME |
| 92 | t-state | FSPMS_Set Input.req ( AREP, Input Data )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Set Input.req ( Input Data ) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 93 | t-state | FSPMS_Get Output.req ( AREP )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Get Output.req ( ) | SAME |
| 94 | t-state | FSPMS_Initiate.rsp(+) ( AREP, Max Len Data Unit, Features Supported, Profile Features Supported, Profile Ident Number, Add Addr Param )<br>/AREP.AR_Type=MS1<br>=><br>MSAC2S_Initiate.rsp(+) ( Res SAP, Max Len Data Unit, Features Supported, Profile Features Supported, Profile Ident Number, Add Addr Param ) | SAME |
| 95 | t-state | FSPMS_Initiate.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Initiate.rsp(-) ( Res SAP, Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 96 | t-state | FSPMS_Read.rsp(+) ( AREP, Length, Data )<br>/AREP.AR_Type=MS1<br>=><br>MSAC1S_Read.rsp(+) ( Length, Data ) | SAME |
| 97 | t-state | FSPMS_Read.rsp(+) ( AREP, Length, Data )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Read.rsp(+) ( Length, Data ) | SAME |
| 98 | t-state | FSPMS_Read.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS1<br>=><br>MSAC1S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 99 | t-state | FSPMS_Read.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 100 | t-state | FSPMS_Write.rsp(+) ( AREP, Length )<br>/AREP.AR_Type=MS1<br>=><br>MSAC1S_Write.rsp(+) ( Length ) | SAME |
| 101 | t-state | FSPMS_Write.rsp(+) ( AREP, Length )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Write.rsp(+) ( Length ) | SAME |
| 102 | t-state | FSPMS_Write.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS1<br>=><br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 103 | t-state | FSPMS_Write.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 104 | t-state | FSPMS_Data Transport.rsp(+) ( AREP, Length, Data )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Data Transport.rsp(+) ( Res SAP, Length, Data ) | SAME |
| 105 | t-state | FSPMS_Data Transport.rsp(-) ( AREP, Error Decode, Error Code 1 Error Code 2 )<br>/AREP.AR_Type=MS2<br>=><br>MSAC2S_Data Transport.rsp(-) ( Res SAP, Error Decode, Error Code 1 Error Code 2 ) | SAME |
| 106 | t-state | FSPMS_Alarm Ack.rsp(+) ( AREP, Slot Number, Alarm Type, Seq Nr )<br>/AREP.AR_Type=MS1<br>=> | SAME |
| 107 | t-state | FSPMS_Alarm Ack.rsp(-) ( AREP, Slot Number, Alarm Type, Seq Nr )<br>/AREP.AR_Type=MS1<br>=> | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 108 | t-state | DMPMS_SInit DLL.cnf ( )<br>=><br>FSPMS DLL Init DP Slave.cnf ( ) | SAME |
| 109 | t-state | MSCY1S_SInit MS0.cnf ( )<br>=><br>AREP=MS0-AR<br>FSPMS_SInit MS0.cnf ( AREP ) | SAME |
| 110 | t-state | MSRM2S_SInit MS2.cnf ( )<br>=><br>FSPMS_SInit MS2.cnf ( ) | SAME |
| 111 | t-state | MSCY1S_Check User Prm Result.cnf(+) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Check User Prm Result.cnf(+) ( AREP ) | SAME |
| 112 | t-state | MSCY1S_Check User Prm Result.cnf(-) ( Status )<br>=><br>AREP=MS0-AR<br>FSPMS_Check User Prm Result.cnf(-) ( AREP, Status ) | SAME |
| 113 | t-state | MSCY1S_Check Cfg Result.cnf(+) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Check Cfg Result.cnf(+) ( AREP ) | SAME |
| 114 | t-state | MSCY1S_Check Cfg Result.cnf(-) ( Status )<br>=><br>AREP=MS0-AR<br>FSPMS_Check Cfg Result.cnf(-) ( AREP, Status ) | SAME |
| 115 | t-state | MSCY1S_Set Cfg.cnf(+) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Cfg.cnf(+) ( AREP ) | SAME |
| 116 | t-state | MSCY1S_Set Cfg.cnf(-) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Cfg.cnf(-) ( AREP ) | SAME |
| 117 | t-state | MSCY1S_Set Input.cnf(+) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Input.cnf(+) ( AREP ) | SAME |
| 118 | t-state | MSCY1S_Set Input.cnf(-) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Input.cnf(-) ( AREP ) | SAME |
| 119 | t-state | MSCY1S_Get Output.cnf(+) ( Output Data, Clear Flag, New Flag )<br>=><br>AREP=MS0-AR<br>FSPMS_Get Output.cnf(+) ( AREP, Output Data, Clear Flag, New Flag ) | SAME |
| 120 | t-state | MSCY1S_Get Output.cnf(-) ( )<br>=><br>AREP=MS0-AR<br>FSPMS_Get Output.cnf(-) ( AREP ) | SAME |
| 121 | t-state | MSCY1S_Set Slave Add.ind ( New Slave Add, Ident Number, No Add Chg, Rem Slave Data )<br>=><br>AREP=MS0-AR<br>FSPMS_Set Slave Add.ind ( AREP, New Slave Add, Ident Number, No Add Chg, Rem Slave Data ) | SAME |
| 122 | t-state | MSCY1S_Check User Prm.ind ( User Prm Data )<br>=><br>AREP=MS0-AR<br>FSPMS_Check User Prm.ind ( AREP, User Prm Data ) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 123 | t-state | MSCY1S_Check Cfg.ind ( Check Cfg Mode, Cfg Data )<br>=><br>AREP=MS0-AR<br>FSPMS_Check Cfg.ind ( AREP, Check Cfg Mode, Cfg Data ) | SAME |
| 124 | t-state | MSCY1S_New Output.ind ( Clear Flag )<br>=><br>AREP=MS0-AR<br>FSPMS_New Output.ind ( AREP, Clear Flag ) | SAME |
| 125 | t-state | MSCY1S_Global Control.ind ( Clear Command, Sync Command, Freeze Command, Group Select )<br>=><br>AREP=MS0-AR<br>FSPMS_Global Control.ind ( AREP, Clear Command, Sync Command, Freeze Command, Group Select ) | SAME |
| 126 | t-state | MSAC2S_Initiate.ind ( Res SAP, Features Supported, Profile Features Supported, Profile Ident Number, Add Addr Param )<br>/SetContext(Res_SAP, Service)=TRUE<br>=><br>FSPMS_Initiate.ind ( AREP, Features Supported, Profile Features Supported, Profile Ident Number, Add Addr Param ) | SAME |
| 127 | t-state | MSAC2S_Abort.ind ( Res SAP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)<br>/SetContext(Res_SAP, Service)=TRUE<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>LR_State[AREP]:=W-LR-IND<br>FSPMS_Abort.ind ( AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) | SAME |
| 128 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/SetContext(Res_SAP, Service)=TRUE &&<br>Index<>255<br>=><br>FSPMS_Read.ind ( AREP, Slot Number, Index, Length ) | SAME |
| 129 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/SetContext(Res_SAP, Service)=TRUE &&<br>Index<>255<br>=><br>FSPMS_Read.ind ( AREP, Slot Number, Index, Length ) | SAME |
| 130 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/SetContext(Res_SAP, Service)=TRUE &&<br>Index<>255<br>=><br>FSPMS_Write.ind ( AREP, Slot Number, Index, Length, Data ) | SAME |
| 131 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/SetContext(Res_SAP, Service)=TRUE &&Index<>255<br>=><br>FSPMS_Write.ind ( AREP, Slot Number, Index, Length, Data ) | SAME |
| 132 | t-state | MSAC2S_Data Transport.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/SetContext(Res_SAP, Service)=TRUE<br>=><br>FSPMS_Data Transport.ind ( AREP, Slot Number, Index, Length, Data ) | SAME |
| 133 | t-state | FSPMS_Get_Publisher_Data.req ( AREP, CREP )<br>/AREP.AR_Type=MS0<br>=><br>Rem_Add=CREP.Rem_Add<br>SSCY1S_Get_Publisher_Data.req (Rem_Add) | SAME |
| 134 | t-state | FSPMS_Start_Subscriber.req ( AREP, CREP )<br>/AREP.AR_Type=MS0<br>=><br>Rem_Add=CREP.Rem_Add<br>SSCY1S_Start_Subscriber.req (Rem_Add) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 135 | t-state | FSPMS_Stop_Subscriber.req ( AREP, CREP )<br>/AREP.AR_Type=MS0<br>=><br>Rem_Add=CREP.Rem_Add<br>SSCY1S_Stop_Subscriber.req (Rem_Add) | SAME |
| 136 | t-state | FSPMS_Check_Ext_User_Prm_Result.req ( AREP, Ext_Prm_OK )<br>/AREP.AR_Type=MS0<br>=><br>MSCY1S_Check_Ext_User_Prm_Result.req (Ext_Prm_OK) | SAME |
| 137 | t-state | SSCY1S_Get_Publisher_Data.cnf(+) (Rem_Add, Data, New_Flag)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Get_Publisher_Data.cnf(+) (AREP, CREP, Data, New_Flag) | SAME |
| 138 | t-state | SSCY1S_Get_Publisher_Data.cnf(-) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Get_Publisher_Data.cnf(-) (AREP, CREP) | SAME |
| 139 | t-state | SSCY1S_Start_Subscriber.cnf(+) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Start_Subscriber.cnf(+) (AREP, CREP) | SAME |
| 140 | t-state | SSCY1S_Start_Subscriber.cnf(-) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Start_Subscriber.cnf(-) (AREP, CREP) | SAME |
| 141 | t-state | SSCY1S_Stop_Subscriber.cnf(+) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Stop_Subscriber.cnf(+) (AREP, CREP) | SAME |
| 142 | t-state | SSCY1S_Stop_Subscriber.cnf(-) (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Stop_Subscriber.cnf(-) (AREP, CREP) | SAME |
| 143 | t-state | SSCY1S_New_Publisher_Data.ind (Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_New_Publisher_Data.ind(AREP, CREP) | SAME |
| 144 | t-state | SSCY1S_Publisher_Active.ind (Rem_Add, Status)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMS_Publisher_Active.ind(AREP, CREP, Status) | SAME |
| 145 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.IL.Extended_Function_Num=Initiate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>Current Load Type[AREP]:=Data.w.IL.Load_Type<br>LR Index:=Data.w.IL.LR_Index<br>Load Type:=Data.w.IL.Load_Type<br>Load Image Size:=Data.w.IL.Load_Image_Size<br>User Specific:=Data.w.IL.User_Specific<br>Intersegment Request Timeout:=Data.w.IL.Intersegment_Request_Timeout<br>FSPMS_Initiate_Load.ind ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>LR_State[AREP]:=W-LR-RES | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 146 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.IL.Extended_Function_Num=Initiate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>Current Load Type[AREP]:=Data.w.IL.Load_Type<br>LR Index:=Data.w.IL.LR_Index<br>Load Type:=Data.w.IL.Load_Type<br>Load Image Size:=Data.w.IL.Load_Image_Size<br>User Specific:=Data.w.IL.User_Specific<br>Intersegment Request Timeout:=Data.w.IL.Intersegment_Request_Timeout<br>FSPMS_Initiate_Load.ind ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 147 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.IL.Extended_Function_Num=Initiate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>CurrentLoadType[AREP]:=Data.w.IL.Load_Type<br>CurrentLRIndex[AREP]:=Data.w.IL.LR_Index<br>LR Index:=Data.w.IL.LR_Index<br>Load Type:=Data.w.IL.Load_Type<br>Load Image Size:=Data.w.IL.Load_Image_Size<br>User Specific:=Data.w.IL.User_Specific<br>Intersegment Request Timeout:=Data.w.IL.Intersegment_Request_Timeout<br>FSPMS_Initiate_Load.ind ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 148 | t-state | FSPMS_Initiate_Load.rsp(+) ( AREP, Actual LR Size, Max Response Delay, Max Segment Length, User Specific )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].LR:=TRUE<br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.IL.Actual_LR_Size:=Actual LR Size<br>CurrentBuffer[AREP].data.r.IL.Max_Response_Delay:=Max Response Delay<br>CurrentBuffer[AREP].data.r.IL.Max_Segment_Length:=Max Segment Length<br>CurrentBuffer[AREP].data.r.IL.User_Specific:=User Specific<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 149 | t-state | FSPMS_Initiate_Load.rsp(+) ( AREP, Actual LR Size, Max Response Delay, Max Segment Length, User Specific )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].LR:=TRUE<br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.IL.Actual_LR_Size:=Actual LR Size<br>CurrentBuffer[AREP].data.r.IL.Max_Response_Delay:=Max Response Delay<br>CurrentBuffer[AREP].data.r.IL.Max_Segment_Length:=Max Segment Length<br>CurrentBuffer[AREP].data.r.IL.User_Specific:=User Specific<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 150 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )/LR_State[AREP]==W-LR-IND&SetContext(Res_SAP, Service)=TRUE &&(Index=255 && CurrentBuffer[AREP].Empty=FALSE && CurrentBuffer[AREP].LR=FALSE )=>Data:=CurrentBuffer[AREP].dataLength:=sizeof(CurrentBuffer[AREP].data)MSAC1S_Read.rsp(+) ( Length, Data )LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 151 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=FALSE &&<br>CurrentBuffer[AREP].LR=TRUE )<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Data:=CurrentBuffer[AREP].data<br>Length:=sizeof(CurrentBuffer[AREP].data)<br>MSAC1S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 152 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=FALSE &&<br>CurrentBuffer[AREP].LR=FALSE )<br>=><br>Data:=CurrentBuffer[AREP].data<br>Length:=sizeof(CurrentBuffer[AREP].data)<br>MSAC2S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 153 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=FALSE &&<br>CurrentBuffer[AREP].LR=TRUE )<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Data:=CurrentBuffer[AREP].data<br>Length:=sizeof(CurrentBuffer[AREP].data)<br>MSAC2S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 154 | t-state | FSPMS_Initiate_Load.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 155 | t-state | FSPMS_Initiate_Load.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 156 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE<br>&&(Index=255 && CurrentBuffer[AREP].Empty=TRUE &&<br>CurrentBuffer[AREP].LR=TRUE )<br>=><br>LR Index:=CurrentLRIndex[AREP]<br>Segment Length:=Length<br>FSPMS_Pull_Segment.ind ( AREP, Slot Number, LR Index, Segment Length )<br>LR_State[AREP]:=W-LR-RES | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 157 | t-state | MSAC1S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=TRUE &&<br>CurrentBuffer[AREP].LR=FALSE )<br>=><br>Error Decode:=DPV1<br>Error Code 1:=state conflict<br>Error Code 2:=0<br>MSAC1S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 158 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=TRUE &&<br>CurrentBuffer[AREP].LR=TRUE )<br>=><br>LR Index:=CurrentLRIndex[AREP]<br>Segment Length:=Length<br>FSPMS_Pull_Segment.ind ( AREP, Slot Number, LR Index, Segment Length )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 159 | t-state | MSAC2S_Read.ind ( Res SAP, Slot Number, Index, Length )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && CurrentBuffer[AREP].Empty=TRUE &&<br>CurrentBuffer[AREP].LR=FALSE )<br>=><br>Error Decode:=DPV1<br>Error Code 1:=state conflict<br>Error Code 2:=0<br>MSAC2S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 160 | t-state | FSPMS_Pull_Segment.rsp(+) ( AREP, Segment Length, Segment Number, More Follows, Data )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Data.PULL.Extended_Function_Num=Pull<br>Data.PULL.Sequence Number:=Segment Number<br>Data.PULL.Options.More Follows:=More Follows<br>Data.PULL.Region Data:= Data<br>Length:=6+Segment Length<br>MSAC1S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 161 | t-state | FSPMS_Pull_Segment.rsp(+) ( AREP, Segment Length, Segment Number, More Follows, Data )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Data.PULL.Extended_Function_Num=Pull<br>Data.PULL.Sequence Number:=Segment Number<br>Data.PULL.Options.More Follows:=More Follows<br>Data.PULL.Region Data:= Data<br>Length:=6+Segment Length<br>MSAC2S_Read.rsp(+) ( Length, Data )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 162 | t-state | FSPMS_Pull Segment.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 163 | t-state | FSPMS_Pull Segment.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC2S_Read.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 164 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.PUSH.Extended_Function_Num=Push)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>LR Index:=Data.PUSH.LR_Index<br>Segment Length:=Length<br>Segment Number:=Data.PUSH.Segment_Number<br>More Follows:=Data.PUSH.More_Follows<br>Data:=Data.PUSH.data<br>FSPMS_Push_Segment.ind ( AREP, Slot Number, LR Index, Segment Length,<br>Segment Number, More Follows, Data )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 165 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.PUSH.Extended_Function_Num=Push)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>LR Index:=Data.PUSH.LR_Index<br>Segment Length:=Length<br>Segment Number:=Data.PUSH.Segment_Number<br>More Follows:=Data.PUSH.More_Follows<br>Data:=Data.PUSH.data<br>FSPMS_Push_Segment.ind ( AREP, Slot Number, LR Index, Segment Length,<br>Segment Number, More Follows, Data )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 166 | t-state | FSPMS_Push_Segment.rsp(+) ( AREP)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 167 | t-state | FSPMS_Push_Segment.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 168 | t-state | FSPMS_Push_Segment.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 169 | t-state | FSPMS_Push_Segment.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 170 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.TL.Extended_Function_Num=Terminate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>LR Index:=Data.w.TL.LR_Index<br>FSPMS_Terminate_Load.ind ( AREP, Slot Number, LR Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 171 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.TL.Extended_Function_Num=Terminate_Load)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>LR Index:=Data.w.TL.LR_Index<br>FSPMS_Terminate_Load.ind ( AREP, Slot Number, LR Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 172 | t-state | FSPMS_Terminate_Load.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].LR:=FALSE<br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r:=NIL<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 173 | t-state | FSPMS_Terminate_Load.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].LR:=FALSE<br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r:=NIL<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 174 | t-state | FSPMS_Terminate_Load.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 175 | t-state | FSPMS_Terminate_Load.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 176 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&CurrentBuffer[AREP].LR:=FALSE &&SetContext(Res_SAP, Service)=TRUE<br>&&(Index=255 && Length >= 4) &&(Data.w.CALL.Extended_Function_Num=CALL)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br><br>FI Index:=Data.w.CALL.FI_Index<br>Entity_Number:=Data.w.CALL. Entity_Number<br>Execution Argument:=Data.w.CALL.Execution_Argument<br>FSPMS_Call.ind ( AREP, Slot Number, Entity_Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 177 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&CurrentBuffer[AREP].LR:=True &&<br>SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.CALL.Extended_Function_Num=CALL)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=state conflict<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 178 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&CurrentBuffer[AREP].LR:=FALSE &&<br>SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.CALL.Extended_Function_Num=CALL)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br><br>Entity_Number:=Data.w.CALL. Entity_Number<br>FI Index:=Data.w.CALL.FI_Index<br>Execution Argument:=Data.w.CALL.Execution_Argument<br>FSPMS_Call.ind ( AREP, Slot Number, Entity_Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 179 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&CurrentBuffer[AREP].LR:=True &&<br>SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.CALL.Extended_Function_Num=CALL)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=state conflict<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 180 | t-state | FSPMS_Call.rsp(+) ( AREP, Result Argument )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.CALL.ResultArgument:=Result Argument<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 181 | t-state | FSPMS_Call.rsp(+) ( AREP, Result Argument )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.CALL.ResultArgument:=Result Argument<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 182 | t-state | FSPMS_Call.rsp(-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 183 | t-state | FSPMS_Call.rsp(-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 184 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Start)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>Execution Argument:=Data.FIS.Execution_Argument<br>FSPMS_Start.ind ( AREP, Slot Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 185 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Start)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>Execution Argument:=Data.FIS.Execution_Argument<br>FSPMS_Start.ind ( AREP, Slot Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 186 | t-state | FSPMS_Start.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 187 | t-state | FSPMS_Start.rsp(+) ( AREP )/LR_State[AREP]==W-LR-RES&AREP.AR_Type=MS2=>Length:=CurrentBuffer[AREP].WriteLengthMSAC2S_Write.rsp(+) ( Length )LR_State[AREP]:=W-LR-IND | SAME |
| 188 | t-state | FSPMS_Start.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 189 | t-state | FSPMS_Start.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 190 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Stop)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>FSPMS_Stop.ind ( AREP, Slot Number, FI Index)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 191 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Stop)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>FSPMS_Stop.ind ( AREP, Slot Number, FI Index)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 192 | t-state | FSPMS_Stop.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 193 | t-state | FSPMS_Stop.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 194 | t-state | FSPMS_Stop.rsp(-) ( AREP, Error Code )/LR_State[AREP]==W-LR-RES&AREP.AR_Type=MS1=>CurrentBuffer[AREP].Empty:=TRUEError Decode:=DPV1Error Code 1:=Error CodeError Code 2:=0MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 195 | t-state | FSPMS_Stop.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 196 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Resume)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>Execution Argument:=Data.FIS.Execution_Argument<br>FSPMS_Resume.ind ( AREP, Slot Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 197 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Resume)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>Execution Argument:=Data.FIS.Execution_Argument<br>FSPMS_Resume.ind ( AREP, Slot Number, FI Index, Execution Argument)<br>LR_State[AREP]:=W-LR-RES | SAME |
| 198 | t-state | FSPMS_Resume.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 199 | t-state | FSPMS_Resume.rsp(+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 200 | t-state | FSPMS_Resume.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 201 | t-state | FSPMS_Resume.rsp(-) ( AREP, Error Code )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 202 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Reset)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>FSPMS_Reset.ind ( AREP, Slot Number, FI Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 203 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.FIS.Extended_Function_Num=Reset)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.FIS.FI_Index<br>FSPMS_Reset.ind ( AREP, Slot Number, FI Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 204 | t-state | FSPMS_Reset.rsp (+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 205 | t-state | FSPMS_Reset.rsp (+) ( AREP )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 206 | t-state | FSPMS_Reset.rsp (-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 207 | t-state | FSPMS_Reset.rsp (-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC2S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 208 | t-state | MSAC1S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.STATE.Extended_Function_Num=Get_State)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.w.STATE.FI_Index<br>FSPMS_Get_FI_State.ind ( AREP, Slot Number, FI Index )<br>LR_State[AREP]:=W-LR-RES | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement / Condition => action | Prochain état |
|---|---|---|---|
| 209 | t-state | MSAC2S_Write.ind ( Res SAP, Slot Number, Index, Length, Data )<br>/LR_State[AREP]==W-LR-IND<br>&SetContext(Res_SAP, Service)=TRUE &&<br>(Index=255 && Length >= 4) &&<br>(Data.w.STATE.Extended_Function_Num=Get_State)<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>CurrentBuffer[AREP].WriteLength:=Length<br>FI Index:=Data.w.STATE.FI_Index<br>FSPMS_Get_FI_State.ind ( AREP, Slot Number, FI Index )<br>LR_State[AREP]:=W-LR-RES | SAME |
| 210 | t-state | FSPMS_Get_FI_State.rsp (+) ( AREP, FI State )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.STATE.FI_State:=FI State<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC1S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 211 | t-state | FSPMS_Get_FI_State.rsp (+) ( AREP, FI State )<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=FALSE<br>CurrentBuffer[AREP].data.r.STATE.FI_State:=FI State<br>Length:=CurrentBuffer[AREP].WriteLength<br>MSAC2S_Write.rsp(+) ( Length )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 212 | t-state | FSPMS_Get_FI_State.rsp(-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS1<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |
| 213 | t-state | FSPMS_Get_FI_State.rsp(-) ( AREP, Error Code)<br>/LR_State[AREP]==W-LR-RES<br>&AREP.AR_Type=MS2<br>=><br>CurrentBuffer[AREP].Empty:=TRUE<br>Error Decode:=DPV1<br>Error Code 1:=Error Code<br>Error Code 2:=0<br>MSAC1S_Write.rsp(-) ( Error Decode, Error Code 1 Error Code 2 )<br>LR_State[AREP]:=W-LR-IND | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 8.1.4 Fonctions

Le Tableau 36 contient les fonctions utilisées par le FSPMS, leurs arguments et leurs descriptions.

**Tableau 36 – Fonctions utilisées par le FSPMS**

| Nom de fonction | Description |
|---|---|
| SetContext(Rem_Add, Service) | Cette fonction vérifie s'il existe une entrée saisie pour les entrées Rem_Add respectivement Res_SAP et le Service invoqué dans l'ARL et la CRL correspondante de l'esclave DP.<br>A) S'il existe une entrée,<br>elle retourne TRUE et règle contexte local selon le CREP et l'AREP courants.<br>B) Autrement, elle retourne FALSE. |
| FILL(Outstanding_Alarm_TABLE.Alarm_Status, State) | Cette fonction met chaque entrée saisie de l'Outstanding_Alarm_TABLE bidimensionnelle à la valeur initiale donnée de "State". Les paramètres valides pour State sont pending/not_pending. La fonction n'a pas de valeur de retour. |
| Search_In_Outstanding_Alarm_TABLE(Alarm_Status) | Cette fonction explore chaque entrée saisie de l'Outstanding_Alarm_TABLE bidimensionnelle pour trouver un Alarm_Status spécifique. La valeur de retour de cette fonction est un indice qui se réfère à une entrée saisie de l'Alarm_Table qui satisfait aux critères d'Alarm_Status pending/not_pending. |
| SReset_Req_Alarm_FIFO() | Cette fonction réinitialise le FIFO pour les alarmes pas encore envoyées. Au cours de cette fonction, toutes les entrées saisies sont effacées La fonction n'a pas de valeur de retour. |
| Store_To_Req_Alarm_FIFO(Alarm) | Au moyen de cette fonction, l'alarme est stockée dans le Req_Alarm_FIFO. La fonction n'a pas de valeur de retour. |
| Load_From_Req_Alarm_FIFO () | Au moyen de cette fonction, une 'alarme est lue dans le Req_Alarm_FIFO. La valeur de retour est une alarme. |
| Req_Alarm_FIFO_state () | Cette fonction vérifie le FIFO pour rechercher des alarmes pas encore envoyées. La fonction a les valeurs de retour empty/not_empty. |
| Acls(Alarm_Type) | Cette fonction calcule l'indice relatif à l'Alarm_Type comme valeur de retour comme suit:<br>si (Alarm_Type = 1) retourner 0<br>si (Alarm_Type = 2) retourner 1<br>si (Alarm_Type = 3) retourner 2<br>si (Alarm_Type = 4) retourner 3<br>si (Alarm_Type = 5) retourner 4<br>si (Alarm_Type = 6) retourner 5<br>si (Alarm_Type >= 32) && (Alarm_Type <= 126) retourner 6 |
| IsDynamicAttributeLoadable(Attribute) | Cette fonction vérifie l'ARL et la CRL si l'attribut donné est chargeable dans l'état courant.<br>Pour l'attribut DXB-Linktable, la fonction retourne TRUE:<br>Si tous les éléments dans la DXB-Linktable ont un attribut correspondant dans la CRL et aucun des diagrammes d'états SSCY1S correspondants n'est dans l'état RUN. |
| SetDynamicAtrribute(Attribute) | Cette fonction stocke l'attribut donné dans l'ARL ou la CRL.<br>Pour l'attribut DXB-Linktable, chaque entrée est stockée dans l'attribut CRL correspondant. |

IECNORM.COM : Click to view the full PDF of 61158-6-3:2010

**Macros**

**ALARM_ENABLED**

(

((Alarm_Type=3 OR Alarm_Type=4) &&

Actual_Enabled_Alarms[7]=TRUE)

OR ((Alarm_Type=2) && Actual_Enabled_Alarms[6]=TRUE)

OR ((Alarm_Type=1) && Actual_Enabled_Alarms[5]=TRUE)

OR ((Alarm_Type>=32 && Alarm_Type<=126)
&& Actual_Enabled_Alarms[4]=TRUE)
OR ((Alarm_Type=5) && Actual_Enabled_Alarms[3]=TRUE)
OR ((Alarm_Type=6) && Actual_Enabled_Alarms[2]=TRUE)
)

## 8.2 FSPMM1

### 8.2.1 Définitions des primitives

#### 8.2.1.1 Primitives échangées entre AP-Context et FSPMM1

Le Tableau 37 montre les primitives de service, y compris leurs paramètres associés, émises par l'AP-Context et reçues par le FSPMM1.

**Tableau 37 – Primitives émises par l'AP-Context vers le FSPMM1**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Init DP master Cl1.req | AP-Context | Bus Para | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3 et dans la CEI 61158-3-3 |
| Reset DP master Cl1.req | AP-Context | (aucun) | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3. |
| Load ARL DP master Cl1.req | AP-Context | Alarm Max,<br>List of ARL Entries | |
| Get ARL DP master Cl1.req | AP-Context | (aucun) | |
| Load CRL DP master Cl1.req | AP-Context | Update,<br>List of CRL Entries | |
| Get CRL DP master Cl1.req | AP-Context | (aucun) | |
| CRL Slave Activate.req | AP-Context | Activate | |
| CRL New Prm.req | AP-Context | New Prm | |
| CRL New Prm Data.req | AP-Context | Prm Data | |
| Abort DP master Cl1.req | AP-Context | AREP | |
| Mark DP master Cl1.req | AP-Context | AREP | |
| Set Mode DP master Cl1.req | AP-Context | AREP,<br>USIF State | |
| Load Bus Par DP master Cl1.req | AP-Context | Bus Para | |
| Delete SC DP master Cl1.req | AP-Context | Adresses | |
| Read Value DP master Cl1.req | AP-Context | Variable | |
| Get Slave Diag.req | AP-Context | AREP,<br>CREP | |
| Set Output.req | AP-Context | AREP,<br>CREP,<br>Slot Number,<br>Output Data,<br>Final | |
| Get Input.req | AP-Context | AREP,<br>CREP,<br>Slot Number | |
| Global Control.req | AP-Context | AREP,<br>Sync Command,<br>Freeze Command,<br>Group Select | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Read.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length | |
| Write.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Alarm Ack.req(+) | AP-Context | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Get Master Diag.rsp(+) | AP-Context | AREP,<br>Diagnosis Data | |
| Get Master Diag.rsp(-) | AP-Context | AREP,<br>Status | |
| Start Seq.rsp(+) | AP-Context | AREP,<br>Max Len Data Unit | |
| Start Seq.rsp(-) | AP-Context | AREP,<br>Status | |
| Download.rsp(+) | AP-Context | AREP | |
| Download.rsp(-) | AP-Context | AREP,<br>Status | |
| Upload.rsp(+) | AP-Context | AREP,<br>Data | |
| Upload.rsp(-) | AP-Context | AREP,<br>Status | |
| End Seq.rsp(+) | AP-Context | AREP | |
| End Seq.rsp(-) | AP-Context | AREP,<br>Status | |
| Act Param.rsp(+) | AP-Context | AREP | |
| Act Param.rsp(-) | AP-Context | AREP,<br>Status | |
| Set Time.req | AP-Context | AREP,<br>Time Value<br>Local Time Diff<br>Summertime<br>Accuracy<br>Synchronisation<br>Active<br>Announcement Hour | |
| Initiate Load.req | AP-Context | AREP,<br>Slot Number,<br>LR Index,<br>Load Type,<br>Load Image Size,<br>Intersegment<br>Request,<br>Timeout<br>User_Specific | |
| Pull Segment.req | AP-Context | AREP,<br>Slot Number,<br>LR Index,<br>Segment Length | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Push Segment.req | AP-Context | AREP,<br>Slot Number<br>LR Index,<br>Segment Length<br>Segment Number,<br>More Follows,<br>Data | |
| Terminate Load.req | AP-Context | AREP<br>Slot Number<br>LR Index | |
| Start.req | AP-Context | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Stop.req | AP-Context | AREP<br>Slot Number<br>FI Index | |
| Resume.req | AP-Context | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Reset.req | AP-Context | AREP<br>Slot Number<br>FI Index | |
| Call.req | FSPMS | AREP<br>Slot Number<br>Entity Number<br>FI Index<br>Execution Argument | |
| Get_FI_State.req | FSPMS | AREP<br>Slot Number<br>FI Index | |

Le Tableau 38 montre les primitives de service, y compris leurs paramètres associés, émises par le FSPMM1 et reçues par l'AP-Context.

**Tableau 38 – Primitives émises par le FSPMM1 vers l'AP-Context**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Init DP master CI1.cnf | FSPMM1 | (aucun) | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3. |
| Reset DP master CI1.cnf | FSPMM1 | (aucun) | |
| Load ARL DP master CI1.cnf(+) | FSPMM1 | (aucun) | |
| Load ARL DP master CI1.cnf(-) | FSPMM1 | Status | |
| Get ARL DP master CI1.cnf(+) | FSPMM1 | Alarm Max,<br>List of ARL Entries | |
| Get ARL DP master CI1.cnf(-) | FSPMM1 | Status | |
| Load CRL DP master CI1.cnf(+) | FSPMM1 | (aucun) | |
| Load CRL DP master CI1.cnf(-) | FSPMM1 | Status | |
| Get CRL DP master CI1.cnf(+) | FSPMM1 | List of CRL Entries | |
| Get CRL DP master CI1.cnf(-) | FSPMM1 | Status | |
| CRL Slave Activate.cnf(+) | FSPMM1 | | |
| CRL Slave Activate.cnf(-) | FSPMM1 | Status | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| CRL New Prm.cnf(+) | FSPMM1 | | |
| CRL New Prm.cnf(-) | FSPMM1 | Status | |
| CRL New Prm Data.cnf(+) | FSPMM1 | | |
| CRL New Prm Data.cnf(-) | FSPMM1 | Status | |
| Mark DP master Cl1.cnf(+) | FSPMM1 | AREP,<br>Dia | |
| Mark DP master Cl1.cnf(-) | FSPMM1 | AREP,<br>Status | |
| Set Mode DP master Cl1.cnf(+) | FSPMM1 | AREP,<br>Bus Accessible | |
| Set Mode DP master Cl1.cnf(-) | FSPMM1 | AREP,<br>Bus Accessible | |
| Load Bus Par DP master Cl1.cnf | FSPMM1 | Status | |
| Delete SC DP master Cl1.cnf | FSPMM1 | Status | |
| Read Value DP master Cl1.cnf | FSPMM1 | Value, Status | |
| Get Slave Diag.cnf(+) | FSPMM1 | AREP,<br>CREP,<br>Diag Data | |
| Get Slave Diag.cnf(-) | FSPMM1 | AREP,<br>CREP | |
| Set Output.cnf(+) | FSPMM1 | AREP,<br>CREP,<br>Slot Number | |
| Set Output.cnf(-) | FSPMM1 | AREP,<br>CREP,<br>Slot Number,<br>Status | |
| Get Input.cnf(+) | FSPMM1 | AREP,<br>CREP,<br>Slot Number,<br>Input Data | |
| Get Input.cnf(-) | FSPMM1 | AREP,<br>CREP,<br>Slot Number,<br>Status | |
| Global Control.cnf(+) | FSPMM1 | AREP | |
| Global Control.cnf(-) | FSPMM1 | AREP,<br>Status | |
| Read.cnf(+) | FSPMM1 | AREP,<br>Length,<br>Data | |
| Read.cnf(-) | FSPMM1 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.cnf(+) | FSPMM1 | AREP,<br>Length | |
| Write.cnf(-) | FSPMM1 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Alarm Ack.cnf(+) | FSPMM1 | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Alarm Ack.cnf(-) | FSPMM1 | AREP<br>Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Get Master Diag.ind | FSPMM1 | AREP,<br>Mdiag Identifier | |
| Start Seq.ind | FSPMM1 | AREP,<br>Area Code,<br>Timeout | |
| Download.ind | FSPMM1 | AREP,<br>Area Code,<br>Add Offset,<br>Data | |
| Upload.ind | FSPMM1 | AREP,<br>Area Code,<br>Add Offset,<br>Data Len | |
| End Seq.ind | FSPMM1 | AREP | |
| Act Param.ind | FSPMM1 | AREP,<br>Area Code,<br>Activate | |
| Act Para Brct.ind | FSPMM1 | AREP,<br>Area Code | |
| New Slave Diag.ind | FSPMM1 | AREP,<br>CREP | |
| New Input.ind | FSPMM1 | AREP | |
| Alarm Notification.ind | FSPMM1 | AREP,<br>Slot Number,<br>Alarm Type,<br>Seq Nr,<br>Add Ack,<br>Alarm Specifier,<br>Alarm Data | |
| DP master Cl1 Mode Changed.ind | FSPMM1 | AREP,<br>USIF State | |
| DP master Cl1 Event.ind | FSPMM1 | Event,<br>Add Info | |
| DP master Cl1 Started.ind | FSPMM1 | AREP | |
| DP master Cl1 Stopped.ind | FSPMM1 | AREP | |
| Abort.ind | FSPMM1 | AREP | |
| DP master Cl1 Reject.ind | FSPMM1 | AREP,<br>Reason | |
| DP master Cl1 Fault.ind | FSPMM1 | ( none ) | |
| SYNCH.ind | FSPMM1 | AREP | |
| SYNCH Delayed.ind | FSPMM1 | AREP,<br>$T_{SH}$ | |
| DX Finished.ind | FSPMM1 | AREP | |
| Set Time.ind | FSPMM1 | AREP,<br>Time Value,<br>Local Time Diff,<br>Summertime,<br>Accuracy,<br>Synchronisation Active,<br>Announcement Hour | |
| Set Time.cnf | FSPMM1 | AREP,<br>Status | |
| Sync Interval Violation.ind | FSPMM1 | AREP | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Initiate Load.cnf(+) | FSPMM1 | AREP,<br>Actual LR Size,<br>Max Response Delay,<br>Max Segment Length<br>User Specific | |
| Initiate Load.cnf(-) | FSPMM1 | AREP,<br>Error Code | |
| Pull Segment.cnf(+) | FSPMM1 | AREP,<br>Segment Length,<br>Segment Number,<br>More Follows,<br>Data | |
| Pull Segment.cnf(-) | FSPMM1 | AREP,<br>Error Code | |
| Push Segment.cnf(+) | FSPMM1 | AREP | |
| Push Segment.cnf(-) | FSPMM1 | AREP,<br>Error Code | |
| Terminate Load.cnf(+) | FSPMM1 | AREP | |
| Terminate Load.cnf(-) | FSPMM1 | AREP,<br>Error Code | |
| Start.cnf(+) | FSPMM1 | AREP | |
| Start.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Stop.cnf(+) | FSPMM1 | AREP | |
| Stop.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Resume.cnf(+) | FSPMM1 | AREP | |
| Resume.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Reset.cnf(+) | FSPMM1 | AREP | |
| Reset.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Call.cnf(+) | FSPMM1 | AREP<br>Result Argument | |
| Call.cnf(-) | FSPMM1 | AREP<br>Error Code | |
| Get FI State.cnf(+) | FSPMM1 | AREP<br>FI State | |
| Get FI State.cnf(-) | FSPMM1 | AREP<br>Error Code | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

#### 8.2.1.2 Paramètres des primitives de FSPMM1

Les paramètres utilisés avec les primitives échangées entre le FSPMM1 et l'AP-Context sont décrits dans "Définition des services de la FAL" (voir la CEI 61158-5-3).

### 8.2.2 Description de diagramme d'états

Ce Diagramme est utilisé pour coordonner les interactions entre l'AP-Context (Contexte d'AP) et les DMPMM1, MSCY1M, MSAC1M, MSAL1M et MMAC1. Sachant que la prise en charge de plusieurs diagrammes d'états pour la communication avec un esclave pris séparément ainsi que les contraintes de synchronisation du fonctionnement de l'esclave exigent une programmation cohérente, cette tâche est accomplie par le FSPMM1 également.

Pour chaque esclave DP, un jeu individuel de diagrammes d'états (MSCY1M, MSAL1M, MSAC1M) sera établi et sera commandé par le Scheduler (programmateur) dans le maître DP

(Classe 1). Le Scheduler sera command par l'AP-Context avec des services spéciaux. Les Slave-Handlers(MSCY1M), c'est-à-dire descripteurs d'esclaves, seront déclenchés séquentiellement à partir du FSPMS-Scheduler pour accomplir les actions nécessaires Le FSPMM1 notifie les modes de fonctionnement CLEAR et OPERATE en intervalles de temps fixes (Dx_Control_Timer) aux esclaves DP. Cela se produit également aux transitions d'états de FSPMM1.

Le Scheduler fournit quatre modes de fonctionnement pour l'utilisateur:

OFFLINE

STOP

CLEAR

OPERATE

**Variables locales**

**Go_AClr**
(Boolean)

Variable d'état qui indique par la valeur True qu'après avoir fini le cycle esclave courant, le FSPMM1 doit mettre de force tous les esclaves DP dans un état de sécurité (Scheduler-State = CLEAR).

**Internal_Go_AClr**
(Boolean)

Seulement après avoir fini le cycle esclave courant, le FSPMS doit aller à l'état CLEAR si Go_AClr est True. Au cours de chaque cycle, la variable d'état Internal_Go_AClr stocke la valeur réelle de Go_AClr. À la fin de chaque cycle, la valeur de l'Internal_Go_AClr est copiée dans Go_AClr.

**Mark_Active**
(Unsigned8)

Indique l'état de la fonction locale Mark (repérer).

Plage:

Les valeurs possibles sont:

0 => Mark n'est pas active,

1 => Mark est active, mais pas encore en exécution,

2 => Mark est active et en exécution.

**Diag_Active**
(Boolean)

Utilisé pour stocker les informations si l'un au moins des esclaves DP n'est pas dans le mode d'échange de données (True). Les informations sont émises vers l'utilisateur avec Mark.cnf.

**UGC_Count**
(Unsigned8)

Des demandes de Global_Control de l'utilisateur sont possibles à tout moment au cours des modes de fonctionnement CLEAR et OPERATE de FSPMM1. Les confirmations de ces demandes seront transmises à l'utilisateur. Le FSPMM1 utilise le service Global_Control pour envoyer des demandes de Global_Control cyclique avec son mode de fonctionnement réel à tous les esclaves DP assignés. Les confirmations des demandes de Global_Control cyclique doivent être utilisées par le FSPMM1. Le FSPMM1 doit savoir quel Global_Control.cnf appartient à quel Global_Control.req appelé précédemment. Il n'est pas possible de résoudre cette relation en utilisant seulement les services de DL. Par conséquent. le FSPMM1 doit

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

obtenir ces informations de relations en utilisant l'ordre séquentiel des confirmations de Global_Control reçues. Les compteurs UGC_Count et CGC_Count sont utilisés pour stocker des informations relatives à l'ordre des demandes et aussi des informations relatives à l'ordre des confirmations.

Avec le compteur UGC_Coun, toutes les demandes de Global_Control d'utilisateur sont comptées par incréments. La valeur initiale est 0. Elle est décrémentée avec chaque Global_Control.cnf. En outre, le compteur est lié au paramètre Bus_Para.Max_User_Global_Control.

Plage: 0.. Bus_Para.Max_User_Global_Control

**CGC_Count**
(Unsigned8)

Avec ce compteur, toutes les demandes de Global_Control cyclique sont comptées lorsqu'elles ont été lancées par le Scheduler. La valeur initiale est 0. Avec chaque nouvelle Global_Control.req cyclique, l'UGC_Count est incrémenté et sa valeur est copiée dans CGC_Count. UGC_Count aussi est décrémenté avec chaque nouvelle Global_Control.cnf. Si ce compteur atteint la valeur 1, la Global_Control.cnf réelle est une confirmation pour une Global_Control.req cyclique

Plage: 0.. Bus_Para.Max_User_Global_Control

**Bus_Accessible**
(Boolean)

Indique l'état réel du système de bus Si des services peuvent être envoyés, sa valeur est True.

**Act_Rem_Add**
(Unsigned8)

Cette variable stocke l'adresse de station réellement utilisée d'un esclave DP.

Plage: 0.. 125

**Act_msac1m_Max_L_sdu_len**
(Unsigned8)

Cette variable est utilisée pour stocker la valeur effective du paramètre msac1m_Max_L_sdu_len.

Plage: 0.. 240

**User_Sched_Reset**
(Boolean)

Cette variable indique si la réinitialisation réelle a été lancée par l'utilisateur (AP-Context) ou par n'importe quel autre diagramme d'états.

**Act_New_Bus_Para**
(Bus_Para)

Cette structure est utilisée pour sauvegarder un jeu réel de paramètres de bus transmis temporairement de l'AP-Context vers le FSPMM1.

**DX_Lock**
(Boolean)

Cette variable repère un cycle actif d'échange de données pour verrouiller le tampon de données et bloquer l'accès utilisateur. La valeur TRUE est mise quand la premier Continue Slave Handler.req est émise jusqu'au DX Finished.ind. La valeur reste FALSE jusqu'à ce que le prochain SYNCH.ind commence de nouveau le cycle d'échange de données.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**DX_Synch**
(Boolean)

Cette variable repère l'intervalle entre une SYNCH.ind et le prochain cycle d'échange de données. La valeur TRUE est mise quand la SYNCH.ind est reçue jusqu'à ce que la première Continue Slave Handler.req soit émise. La valeur reste FALSE jusqu'à la prochaine SYNCH.ind.

**Chg Buffer**
(Boolean)

La valeur TRUE indique que les tampons pour les données d'entrée et de sortie en dessous doivent être changés avec la prochaine DX Finished.ind. Elle reste FALSE jusqu'à la réception de la prochaine Set Output.req (Final=TRUE).

**Bfr_Set_Output**
(Octet String)

Cette variable est un tampon local pour données de sortie qui appartiennent à l'application. Elle permet un accès asynchrone du processus d'application aux données de sortie dans le mode Buffered Synchronised. Le contenu du tampon est rafraîchi par le FSPMM1 lorsque le processus d'application a produit une Set Output.req (Final = TRUE) et la DX Finished.ind a été reçue. Le tampon peut être accessible de nouveau après la prochaine SYNCH.ind.

**Bfr_Get_Input**
(Octet String)

Cette variable est un tampon local pour données d'entrée qui appartiennent à l'application. Elle permet un accès asynchrone du processus d'application aux données d'entrée dans le mode Buffered Synchronised. Le rafraîchissement est effectué ensemble avec le Bfr_Set_Output.

**Set_Output**
(Octet String)

Cette variable est un tampon local pour données de sortie qui appartiennent au réseau. Elle permet un accès asynchrone du processus d'application aux données de sortie dans le mode Buffered Synchronised. Le contenu du tampon est rafraîchi par le FSPMM1 lorsque le processus d'application a produit une Set Output.req (Final = TRUE) et la DX Finished.ind a été reçue. Le tampon peut être accessible de nouveau après la prochaine SYNCH.ind.

**Get_Input**
(Octet String)

Cette variable est un tampon local pour données d'entrée qui appartiennent au réseau. Elle permet un accès asynchrone du processus d'application aux données d'entrée dans le mode Buffered Synchronised. Le rafraîchissement est effectué ensemble avec le Bfr_Set_Output.

**LR_State**
(Array of Unsigned8)

Cette variable stocke l'état de la séquence Load Region du CREP correspondant.

**CurrentExtendedFunctionNum**
(Array of Unsigned 8)

Cette variable stocke le numéro de fonction étendu du service LR actuellement traité du CREP correspondant.

**CurrentSlotNumber**
(Array of Unsigned8)

Cette variable stocke le numéro de position du service LR actuellement traité du CREP correspondant.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Timers**

**Min_SI_Interval_Timer**
Ce temporisateur surveille le temps minimal de contrôle d'accès des esclaves DP. Il est évalué uniquement dans les états CCLEAR et COPERATE du FSPMM1.

**Dx_Control_Interval_Timer**
Si ce temporisateur expire, le service de diffusion Global_Control est exécuté afin d'informer le esclave DP si le maître DP (Classe 1) est dans l'état CLEAR ou OPERATE. Si le temporisateur expire et la confirmation du dernier Global_Control n'a pas encore été reçue, il s'est produit un problème d'accès bus physique.

**Macros**

**INDICATE_BUS_ACCESSIBILITY**
If (Accessible = True) Bus_Accessible = True

### 8.2.3 Table d'états de FSPMM1

Le Tableau 39 contient la description complète du diagramme d'états FSPMM1.

La définition suivante pour un ensemble d'états est utilisée dans le Tableau 39.

t_state = STOP ou CLEAR ou OPERATE

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 39 – Table d'états de FSPMM1**

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 1 | POWER-ON | Load ARL DP Master Cl1.req (Alarm Max, List of ARL Entries)<br>/paramètres valides<br>=><br>Load ARL DP Master Cl1.cnf(+) | POWER-ON |
| 2 | POWER-ON | Load ARL DP Master Cl1.req (Alarm Max, List of ARL Entries)<br>/paramètres non valides<br>=><br>Status:= NO<br>Load ARL DP Master Cl1.cnf(-) (Status) | POWER-ON |
| 3 | POWER-ON | Get ARL DP Master Cl1.req<br>/ARL loaded<br>=><br>Get ARL DP Master Cl1.cnf(+) ( Alarm Max, List of ARL Entries ) | POWER-ON |
| 4 | POWER-ON | Get ARL DP Master Cl1.req<br>/ARL not loaded<br>=><br>Status:= NO<br>Get ARL DP Master Cl1.cnf(-) (Status) | POWER-ON |
| 5 | POWER-ON | Load CRL DP Master Cl1.req ( List of CRL Entries )<br>/paramètres valides<br>=><br>Load CRL DP Master Cl1.cnf(+) | POWER-ON |
| 6 | POWER-ON | Load CRL DP Master Cl1.req ( List of CRL Entries )<br>/paramètres non valides<br>=><br>Status:= NO<br>Load CRL DP Master Cl1.cnf(-) (Status) | POWER-ON |
| 7 | POWER-ON | Get CRL DP Master Cl1.req<br>/CRL loaded<br>=><br>Get CRL DP Master Cl1.cnf(+) (List of CRL Entries) | POWER-ON |
| 8 | POWER-ON | Get CRL DP Master Cl1.req<br>/CRL not loaded<br>=><br>Status:= NO<br>Get CRL DP Master Cl1.cnf(-) (Status) | POWER-ON |
| 9 | POWER-ON | FSPMM1_Init DP Master Cl1.req(Bus_Para)<br>=><br>User_Init:=TRUE<br>USER_ChangedCritical_Para:= FALSE<br>DX_Lock:= FALSE<br>Chg_Buffer:= FALSE<br>Current Extended Function Num[CREP]:=No_Service<br>LR_State[CREP]:=W-LR-REQ<br>DMPMM1_Minit_DLL.req(Bus_Para) | INIT-DMPM |
| 10 | INIT-DMPM | DMPMM1_Minit_DLL.cnf<br>=><br>Act_Rem_Add:= 0<br>MSAC1M_MInit_MSAC1.req(Rem_Add:=Act_Rem_Add) | INIT-AC1 |
| 11 | — | — | — |
| 12 | INIT-AC1 | MSAC1M_MInit_MSAC1.cnf(Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add:= Act_Rem_Add + 1<br>MSAC1M_MInit_MSAC1.req(Rem_Add:=Act_Rem_Add) | INIT-AC1 |
| 13 | INIT-AC1 | MSAC1M_MInit_MSAC1.cnf(Rem_Add)<br>/Act_Rem_Add=125<br>=><br>Act_Rem_Add:= 0<br>MSAL1M_MInit.req(Rem_Add:=Act_Rem_Add) | INIT-AL1 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 14 | INIT-AL1 | MSAL1M_MInit.cnf(Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add:= Act_Rem_Add + 1<br>MSAL1M_MInit.req(Rem_Add:=Act_Rem_Add) | INIT-AL1 |
| 15 | INIT-AL1 | MSAL1M_MInit.cnf(Rem_Add)<br>/Act_Rem_Add = 125<br>=><br>Act_Rem_Add:= 0<br>MSCY1M_Minit_MS0.req(Rem_Add:=Act_Rem_Add) | INIT-CY1 |
| 16 | INIT-CY1 | MSCY1M_Minit_MS0.cnf(Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add:= Act_Rem_Add + 1<br>MSCY1M_Minit_MS0.req(Rem_Add:=Act_Rem_Add) | INIT-CY1 |
| 17 | INIT-CY1 | MSCY1M_Minit_MS0.cnf(Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MM)=TRUE<br>=><br>MMAC1_Minit_MM.req | INIT-MM1 |
| 18 | INIT-CY1 | MSCY1M_Minit_MS0.cnf(Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MM)=FALSE &&<br>USER_ChangedCritical_Para = FALSE<br>=><br>Bus_accessible:= FALSE<br>if(USER_SetMode_Offline=FALSE)<br>FSPMM1_Init DP Master CI1.cnf<br>else<br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible)<br>USER_SetMode_Offline:=FALSE<br>endif | OFFLINE |
| 19 | INIT-CY1 | MSCY1M_Minit_MS0.cnf(Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MM)=FALSE &&<br>USER_ChangedCritical_Para = TRUE<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para:=Act_New_Bus_Para) | LDBP |
| 20 | INIT-MM1 | MMAC1_Minit_MM.cnf<br>/USER_ChangedCritical_Para = TRUE<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para:=Act_New_Bus_Para) | LDBP |
| 21 | INIT-MM1 | MMAC1_Minit_MM.cnf<br>/USER_ChangedCritical_Para = FALSE<br>=><br>Bus_accessible:= FALSE<br>if(USER_SetMode_Offline=FALSE)<br>FSPMM1_Init DP Master CI1.cnf<br>else<br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible)<br>USER_SetMode_Offline:=FALSE<br>endif | OFFLINE |
| 22 | OFFLINE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State <> Stop && USIF_State <> Offline<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(-)(AREP, Bus_accessible) | OFFLINE |
| 23 | OFFLINE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Offline<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible) | OFFLINE |
| 24 | OFFLINE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop && Bus_Para invalid<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(-)(AREP, Bus_accessible) | OFFLINE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 25 | OFFLINE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop && Bus_Para valid<br>=><br>Mark_Active:= 0<br>Diag_Active:= False<br>DMPMM1_Set_Bus_Par.req(Bus_Para) | LDBP |
| 26 | OFFLINE | FSPMM1_Mark.req(AREP)<br>=><br>FSPMM1_Mark.cnf(-)(AREP, Status:=NO) | OFFLINE |
| 27 | OFFLINE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | OFFLINE |
| 28 | OFFLINE | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=NO) | OFFLINE |
| 29 | OFFLINE | FSPMM1_Delete_SC DP Master CI1.req(Address)<br>=><br>DMPMM1_Delete_SC.req(Address) | DELSC-OFF |
| 30 | OFFLINE | FSPMM1_Read_Value DP Master CI1.req(Variable)<br>=><br>DMPMM1_Read_Value.req(Variable) | RDVAL-OFF |
| 31 | LDBP | DMPMM1_Set_Bus_Par.cnf<br>/USER_ChangeCritical_Para = FALSE<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible) | STOP |
| 32 | LDBP | DMPMM1_Set_Bus_Par.cnf<br>/USER_ChangeCritical_Para = TRUE<br>=><br>FSPMM1_Load_Bus_Para.cnf(+) | STOP |
| 33 | DELSC-OFF | DMPMM1_Delete_SC.cnf<br>=><br>FSPMM1_Delete_SC DP Master CI1.cnf | OFFLINE |
| 34 | RDVAL-OFF | DMPMM1_Read_Value.cnf(Status,Value)<br>=><br>FSPMM1_Read_Value DP Master CI1.cnf(Status, Value) | OFFLINE |
| 35 | STOP | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Offline<br>=><br>USER_SetMode_Offline:=TRUE | RESET |
| 36 | STOP | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible) | STOP |
| 37 | STOP | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(-)(AREP, Bus_accessible) | STOP |
| 38 | STOP | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Clear<br>=><br>UGC_Count:= 0<br>CGC_Count:= 0<br>Bus_Acessible:= True<br>Go_AClr:= Bus_Para.Error_Action_Flag<br>Internal_Go_AClr:= False<br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>MSCY1M_Start_Slave_Handler.req(0 .. 125) | CLEAR |
| 39 | STOP | FSPMM1_Mark.req(AREP)<br>=><br>FSPMM1_Mark.cnf(-)(AREP, Status:=NO) | STOP |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 40 | STOP | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | STOP |
| 41 | STOP | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/Bus_Para invalid<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=IV) | STOP |
| 42 | STOP | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters unchanged)<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para) | LDBP-ST-CPU |
| 43 | STOP | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters changed)<br>=><br>Act_New_Bus_Para:= Bus_Para<br>User_ChangedCritical_Para:= VRAI | RESET |
| 44 | STOP | FSPMM1_Delete_SC DP Master CI1.req(Address)<br>=><br>DMPMM1_Delete_SC.req(Address) | DELSC-ST |
| 45 | STOP | FSPMM1_Read_Value DP Master CI1.req(Variable)<br>=><br>DMPMM1_Read_Value.req(Variable) | RDVAL-ST |
| 46 | LDBP-ST-CPU | DMPMM1_Set_Bus_Par.cnf<br>=><br>FSPMM1_Load_Bus_Para.cnf(+) | STOP |
| 47 | DELSC-ST | DMPMM1_Delete_SC.cnf<br>=><br>FSPMM1_Delete_SC DP Master CI1.cnf | STOP |
| 48 | RDVAL-ST | DMPMM1_Read_Value.cnf(Status,Value)<br>=><br>FSPMM1_Read_Value DP Master CI1.cnf(Status, Value) | STOP |
| 49 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Offline<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(-)(AREP, Bus_accessible) | CLEAR |
| 50 | CLEAR | FSPMM1_Mark DP Master CI1.req(AREP)<br>/Mark_Active = 0<br>=><br>Mark_Active:= 1 | CLEAR |
| 51 | CLEAR | FSPMM1_Mark DP Master CI1.req(AREP)<br>/Mark_Active <> 0<br>=><br>FSPMM1_Mark.cnf(-)(AREP, Status:=NO) | CLEAR |
| 52 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop && Mark_Active = 0<br>=><br>Stop Dx_Control_Interval_Timer<br>MSCY1M_Stop_Slave_Handler.req(0..125) | SLHSTP |
| 53 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Stop && Mark_Active <> 0<br>=><br>Mark_Active:= 0<br>Stop Dx_Control_Interval_Timer<br>FSPMM1_Mark.cnf(-)(AREP; Status:=NO)<br>MSCY1M_Stop_Slave_Handler.req(0..125) | SLHSTP |
| 54 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Clear<br>=><br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible) | CLEAR |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 55 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate &&<br>Go_AClr = False && Internal_Go_AClr = False &&<br>CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2 | SGC-CO-WGC |
| 56 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate &&<br>Go_AClr = False && Internal_Go_AClr = False &&<br>CGC_Count = 0<br>=><br>UGC_Count:= UGC_Count+1<br>CGC_Count:= UGC_Count<br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=0,<br>Group_Select:=0) | SGC-CO |
| 57 | CLEAR | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate &&<br>(Go_AClr = True \|\| Internal_Go_AClr = True)<br>=><br>FSPMM1_Set_Mode DP Master Cl1.cnf(-)(AREP, Bus_accessible) | CLEAR |
| 58 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/NOT received all MSCY1M_Start_Slave_Handler.cnf<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | CLEAR |
| 59 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count = 0 && UGC_Count >=<br>Bus_Para.Max_User_Global_Control<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | CLEAR |
| 60 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count = 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control &&<br>(Isochronous Mode = Not Synchronized \|\|<br>Group_Select < 0x80)<br>=><br>UGC_Count:= UGC_Count+1<br>DMPMM1_Global_Control.req(Rem_Add:=CREP.Rem_Add,<br>Control_Command.1:=1,<br>Group_Select) | CLEAR |
| 61 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count > 0 && UGC_Count >=<br>Bus_Para.Max_User_Global_Control+1<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | CLEAR |
| 62 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command,<br>Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count > 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control+1 &&<br>(Isochronous Mode = Not Synchronized \|\|<br>Group_Select < 0x80)<br>=><br>UGC_Count:= UGC_Count+1<br>DMPMM1_Global_Control.req(Rem_Add:=CREP.Rem_Add,<br>Control_Command.1:=1,<br>Group_Select) | CLEAR |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 63 | CLEAR | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 0<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | CLEAR |
| 64 | CLEAR | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= 0 | CLEAR |
| 65 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count = 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control &&<br>Isochronous Mode = (Buffered Synchronized \|\| Enhanced Synchronized) &&<br>Group_Select >= 0x80<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=GE) | CLEAR |
| 66 | CLEAR | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/received all MSCY1M_Start_Slave_Handler.cnf &&<br>CGC_Count > 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control+1 &&<br>Isochronous Mode = (Buffered Synchronized \|\| Enhanced Synchronized) &&<br>Group_Select >= 0x80<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=GE) | CLEAR |
| 67 | CLEAR | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | CLEAR |
| 68 | CLEAR | MSCY1M_Start_Slave_Handler.cnf(Rem_Add)<br>/NOT received all MSCY1M_Start_Slave_Handler.cnf<br>=><br>ignore | CLEAR |
| 69 | CLEAR | MSCY1M_Start_Slave_Handler.cnf(Rem_Add)<br>/received all MSCY1M_Start_Slave_Handler.cnf && Isochronous Mode = Not Synchronized<br>=><br>Internal_Go_AClr:= False<br>Start Min_SI_Interval_Timer(Bus_Para.Min_Slave_Interval)<br>Set_Mode.cnf(Status:=OK, Bus_Accessible)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |
| 70 | CLEAR | MSCY1M_Start_Slave_Handler.cnf(Rem_Add)<br>/received all MSCY1M_Start_Slave_Handler.cnf && Isochronous Mode <> Not Synchronized<br>=><br>Internal_Go_AClr:= False<br>Set_Mode.cnf(Status:=OK, Bus_Accessible)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |
| 71 | CLEAR | DMPMM1_SYNCH.ind<br>=><br>FSPMM1_DP Master Cl1 Fault.ind | RESET |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 72 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/NOT received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active = 2<br>=><br>Diag_Active:= Diag_Active OR Diag<br>Internal_Go_AClr:= Internal_Go_AClr OR<br>(Bus_Para.Error_Action_Flag<br>AND NOT No_AClr) | CLEAR |
| 73 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/NOT received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active <> 2<br>=><br>Internal_Go_AClr:= Internal_Go_AClr OR<br>(Bus_Para.Error_Action_Flag<br>AND NOT No_AClr) | CLEAR |
| 74 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active = 0<br>=><br>Internal_Go_AClr:= Internal_Go_AClr OR (Bus_Para.Error_Action_Flag AND NOT No_AClr)<br>DX_Lock:= FAUX | CCLEAR |
| 75 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active = 1<br>=><br>Mark_Active:=2<br>Diag_Active:= False<br>Internal_Go_AClr:= Internal_Go_AClr OR (Bus_Para.Error_Action_Flag AND NOT No_AClr)<br>DX_Lock:= FAUX | CCLEAR |
| 76 | CLEAR | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf<br>&& Mark_Active = 2<br>=><br>Diag_Active:= Diag_Active OR Diag<br>Internal_Go_AClr:= Internal_Go_AClr OR (Bus_Para.Error_Action_Flag AND NOT No_AClr)<br>Mark_Active:=0<br>DX_Lock:= FALSE<br>FSPMM1_Mark.cnf(+)(AREP, Dia:=Diag_Active) | CCLEAR |
| 77 | CLEAR | Dx_Control_Interval_Timer expired<br>/CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>UGC_Count:= UGC_Count+1<br>CGC_Count:= UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2,<br>Group_Select:=0) | CLEAR |
| 78 | CLEAR | Dx_Control_Interval_Timer expired<br>/CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible:= False | CLEAR |
| 79 | CLEAR | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/Bus_Para invalid<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=IV) | CLEAR |
| 80 | CLEAR | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters changed)<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=NO) | CLEAR |
| 81 | CLEAR | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters unchanged)<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para) | LDBP-CL-CPU |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 82 | CLEAR | FSPMM1_Delete_SC DP Master CI1.req(Address)<br>=><br>DMPMM1_Delete_SC.req(Address) | DELSC-CL |
| 83 | CLEAR | FSPMM1_Read_Value DP Master CI1.req(Variable)<br>=><br>DMPMM1_Read_Value.req(Variable) | RDVAL-CL |
| 84 | SLHSTP | MSCY1M_Stop_Slave_Handler.cnf(Rem_Add)<br>/NOT received all MSCY1M_Stop_Slave_Handler.cnf<br>=><br>ignore | SLHSTP |
| 85 | SLHSTP | MSCY1M_Stop_Slave_Handler.cnf(Rem_Add)<br>/received all MSCY1M_Stop_Slave_Handler.cnf<br>=><br>Set_Mode.cnf(Status:=OK, Bus_Accessible) | STOP |
| 86 | SGC-CO-WGC | Dx_Control_Interval_Timer expired<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible:= False<br>FSPMM1_Set_Mode DP Master CI1.cnf(-)(AREP, Bus_accessible) | CLEAR |
| 87 | SGC-CO-WGC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | SGC-CO-WGC |
| 88 | SGC-CO-WGC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>CGC_Count:= UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=0,<br>Group_Select:=0) | SGC-CO |
| 89 | SGC-CO | Dx_Control_Interval_Timer expired<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible:= False<br>Set_Mode.cnf(Status:=OK, Bus_Accessible) | OPERATE |
| 90 | SGC-CO | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | SGC-CO |
| 91 | SGC-CO | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= 0<br>FSPMM1_Set_Mode DP Master CI1.cnf(+)(AREP, Bus_accessible) | OPERATE |
| 92 | CCLEAR | Min_SI_Interval_Timer expired<br>=><br>Go_AClr:= Internal_Go_AClr<br>Internal_Go_AClr:= False<br>Start Min_SI_Interval_Timer(Bus_Para.Min_Slave_Interval)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 93 | CCLEAR | DMPMM1_SYNCH.ind<br>=><br>DX_Lock:= TRUE<br>Output_Data = Set_Output<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=False)<br>MSCY1M_Set_Output.req(Rem_Add:=0..125, Slot_Number:=ALL, Output_Data)<br>MSCY1M_Get_Input.req(Rem_Add:=0..125, Slot_Number:=ALL)<br>FSPMM1_SYNCH.ind (AREP) | CLEAR |
| 94 | LDBP-CL-CPU | DMPMM1_Set_Bus_Par.cnf<br>=><br>FSPMM1_Load_Bus_Para.cnf(+) | CLEAR |
| 95 | DELSC-CL | DMPMM1_Delete_SC.cnf<br>=><br>FSPMM1_Delete_SC DP Master Cl1.cnf | CLEAR |
| 96 | RDVAL-CL | DMPMM1_Read_Value.cnf(Status,Value)<br>=><br>FSPMM1_Read_Value DP Master Cl1.cnf(Status, Value) | CLEAR |
| 97 | OPERATE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State <> Clear && USIF_State <> Operate<br>=><br>FSPMM1_Set_Mode DP Master Cl1.cnf(-)(AREP, Bus_accessible) | OPERATE |
| 98 | OPERATE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Operate<br>=><br>FSPMM1_Set_Mode DP Master Cl1.cnf(+)(AREP, Bus_accessible) | OPERATE |
| 99 | OPERATE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Clear && CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2) | SGC-OC-WGC |
| 100 | OPERATE | FSPMM1_Set_Mode.req(USIF_State)<br>/USIF_State = Clear && CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2<br>UGC_Count:= UGC_Count+1<br>CGC_Count:= UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2,<br>Group_Select:=0) | SGC-OC |
| 101 | OPERATE | FSPMM1_Mark DP Master Cl1.req(AREP)<br>/Mark_Active = 0<br>=><br>Mark_Active:= 1 | OPERATE |
| 102 | OPERATE | FSPMM1_Mark DP Master Cl1.req(AREP)<br>/Mark_Active <> 0<br>=><br>FSPMM1_Mark.cnf(-)(AREP, Status:=NO) | OPERATE |
| 103 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 1<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | OPERATE |
| 104 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count = 0 && UGC_Count >=<br>Bus_Para.Max_User_Global_Control<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | OPERATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 105 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count = 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control &&<br>(Isochronous Mode = Not Synchronized ||<br>Group_Select < 0x80)<br>=><br>UGC_Count:= UGC_Count+1<br>DMPMM1_Global_Control.req(Rem_Add:=CREP.Rem_Add,<br>Control_Command.1:=0,<br>Group_Select) | OPERATE |
| 106 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count > 0 && UGC_Count >=<br>Bus_Para.Max_User_Global_Control+1<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=NO) | OPERATE |
| 107 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count > 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control+1 &&<br>(Isochronous Mode = Not Synchronized ||<br>Group_Select < 0x80)<br>=><br>UGC_Count:= UGC_Count+1<br>DMPMM1_Global_Control.req(Rem_Add:=CREP.Rem_Add,<br>Control_Command.1:=0,<br>Group_Select) | OPERATE |
| 108 | OPERATE | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 0<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | OPERATE |
| 109 | OPERATE | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= 0 | OPERATE |
| 110 | OPERATE | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | OPERATE |
| 111 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count = 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control &&<br>Isochronous Mode = (Buffered Synchronized \|\| Enhanced Synchronized) &&<br>Group_Select >= 0x80<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=GE) | OPERATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 112 | OPERATE | FSPMM1_Global_Control.req(AREP, Sync_Command, Freeze_Command, Group_Select)<br>/Control_Command.1 = 0 &&<br>CGC_Count > 0 && UGC_Count <<br>Bus_Para.Max_User_Global_Control+1 &&<br>Isochronous Mode = (Buffered Synchronized \|\| Enhanced Synchronized) &&<br>Group_Select >= 0x80<br>=><br>FSPMM1_Global_Control.cnf(-)(AREP, Status:=GE) | OPERATE |
| 113 | OPERATE | DMPMM1_SYNCH.ind<br>=><br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 114 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/NOT received all MSCY1M_Cont_Slave_Handler.cnf &&<br>Mark_Active = 2<br>=><br>Diag_Active:=Diag_Active OR Diag<br>Go_AClr:= Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr) | OPERATE |
| 115 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/NOT received all MSCY1M_Cont_Slave_Handler.cnf &&<br>Mark_Active <> 2<br>=><br>Go_AClr:= Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr) | OPERATE |
| 116 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 0 &&<br>(Isochronous Mode = Not Synchronized \|\| Isochronous Mode = Enhanced Synchronized)<br>=><br>Go_AClr:= Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>DX_Lock:= FALSE<br>FSPMM1_DX_Finished.ind (AREP) | COPERATE |
| 117 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 1 &&<br>(Isochronous Mode = Not Synchronized \|\| Isochronous Mode = Enhanced Synchronized)<br>=><br>Mark_Active:=2<br>Diag_Active:= False<br>Go_AClr:= Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>DX_Lock:= FALSE<br>FSPMM1_DX_Finished.ind (AREP) | COPERATE |
| 118 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 2 &&<br>(Isochronous Mode = Not Synchronized \|\| Isochronous Mode = Enhanced Synchronized)<br>=><br>Diag_Active:=Diag_Active OR Diag<br>Go_AClr:= Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>Mark_Active:=0<br>DX_Lock:= FALSE<br>FSPMM1_Mark.cnf(+)(AREP, Dia:=Diag_Active)<br>DX_Finished.ind (AREP) | COPERATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 119 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 0 &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Go_AClr:= Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>DX_Lock:= FALSE<br>Chg_Bfr (Bfr_Get_Input, Get_Input)<br>FSPMM1_DX_Finished.ind (AREP) | COPERATE |
| 120 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 1 &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Mark_Active:=2<br>Diag_Active:= False<br>Go_AClr:= Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>DX_Lock:= FALSE<br>Chg_Bfr (Bfr_Get_Input, Get_Input)<br>FSPMM1_DX_Finished.ind (AREP) | COPERATE |
| 121 | OPERATE | MSCY1M_Cont_Slave_Handler.cnf(Rem_Add, Diag, No_AClr)<br>/received all MSCY1M_Cont_Slave_Handler.cnf && Mark_Active = 2 &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Diag_Active:=Diag_Active OR Diag<br>Go_AClr:= Go_AClr OR<br>(Bus_Para.Error_Action_Flag AND<br>NOT No_AClr)<br>Mark_Active:=0<br>DX_Lock:= FALSE<br>FSPMM1_Mark.cnf(+)(AREP, Dia:=Diag_Active)<br>DX_Finished.ind (AREP) | COPERATE |
| 122 | OPERATE | Dx_Control_Interval_Timer expired<br>/CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>UGC_Count:= UGC_Count+1<br>CGC_Count:= UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=0,<br>Group_Select:=0) | OPERATE |
| 123 | OPERATE | Dx_Control_Interval_Timer expired<br>/CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_accessible:= False | OPERATE |
| 124 | OPERATE | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/Bus_Para invalid<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=IV) | OPERATE |
| 125 | OPERATE | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters changed)<br>=><br>FSPMM1_Load_Bus_Par.cnf(-)(Status:=NO) | OPERATE |
| 126 | OPERATE | FSPMM1_Load_Bus_Par DP Master CI1.req(Bus_Para)<br>/(Bus_Para valid) && (critical parameters unchanged)<br>=><br>DMPMM1_Set_Bus_Par.req(Bus_Para) | LDBP-OP-CPU |
| 127 | OPERATE | FSPMM1_Delete_SC DP Master CI1.req(Address)<br>=><br>DMPMM1_Delete_SC.req(Address) | DELSC-OP |
| 128 | OPERATE | FSPMM1_Read_Value DP Master CI1.req(Variable)<br>=><br>DMPMM1_Read_Value.req(Variable) | RDVAL-OP |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 129 | SGC-OC-WGC | Dx_Control_Interval_Timer expired<br>/Go_AClr = False<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible:= False<br>FSPMM1_Set_Mode DP Master Cl1.cnf(-)(AREP, Bus_accessible) | OPERATE |
| 130 | SGC-OC-WGC | Dx_Control_Interval_Timer expired<br>/Go_AClr = True<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible:= False | SGC-OC-WGC |
| 131 | SGC-OC-WGC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | SGC-OC-WGC |
| 132 | SGC-OC-WGC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>CGC_Count:= UGC_Count<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2,<br>Group_Select:=0) | SGC-OC |
| 133 | SGC-OC | Dx_Control_Interval_Timer expired<br>/Go_AClr = False<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible:= False<br>Set_Mode.cnf(Status:=OK, Bus_Accessible) | CLEAR |
| 134 | SGC-OC | Dx_Control_Interval_Timer expired<br>/Go_AClr = True<br>=><br>Start Dx_Control_Interval_Timer(Bus_Para.Data_Control_Time/2)<br>Bus_Accessible:= False<br>FSPMM1_DP Master Cl1 Mode_Changed.ind(AREP, USIF_State:=Clear)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |
| 135 | SGC-OC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count > 1<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= CGC_Count-1<br>FSPMM1_Global_Control.cnf(+)(AREP) | SGC-OC |
| 136 | SGC-OC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1 && Go_AClr = False<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= 0<br>FSPMM1_Set_Mode DP Master Cl1.cnf(+)(AREP, Bus_accessible) | CLEAR |
| 137 | SGC-OC | DMPMM1_Global_Control.cnf(Rem_Add, Status)<br>/CGC_Count = 1 && Go_AClr = True<br>=><br>INDICATE_BUS_ACCESSIBILITY<br>UGC_Count:= UGC_Count-1<br>CGC_Count:= 0<br>FSPMM1_DP Master Cl1 Mode_Changed.ind(AREP, USIF_State:=Clear)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=True) | CLEAR |
| 138 | COPERATE | Min_SI_Interval_Timer expired<br>/Go_AClr = False<br>=><br>Start Min_SI_Interval_Timer (Bus_Para.Min_Slave_Interval)<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=False) | OPERATE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 139 | COPERATE | Min_SI_Interval_Timer expired<br>/Go_AClr = True && CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer (Bus_Para.Data_Control_Time/2)<br>Start Min_SI_Interval_Timer (Bus_Para.Min_Slave_Interval) | SGC-OC-WGC |
| 140 | COPERATE | Min_SI_Interval_Timer expired<br>/Go_AClr = True && CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer (Bus_Para.Data_Control_Time/2)<br>UGC_Count:= UGC_Count+1<br>CGC_Count:= UGC_Count<br>Start Min_SI_Interval_Timer (Bus_Para.Min_Slave_Interval)<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2,<br>Group_Select:=0) | SGC-OC |
| 141 | COPERATE | DMPMM1_SYNCH.ind<br>/Go_AClr = False<br>=><br>DX_Lock:= TRUE<br>Output_Data = Set_Output<br>MSCY1M_Cont_Slave_Handler.req(0..125, Output_Clear:=False)<br>MSCY1M_Set_Output.req(Rem_Add:=0..125, Slot_Number:=ALL, Output_Data)<br>MSCY1M_Get_Input.req(Rem_Add:=0..125, Slot_Number:=ALL)<br>FSPMM1_SYNCH.ind (AREP) | COPERATE |
| 142 | COPERATE | DMPMM1_SYNCH.ind<br>/Go_AClr = True && CGC_Count > 0<br>=><br>Start Dx_Control_Interval_Timer (Bus_Para.Data_Control_Time/2)<br>DX_Lock:= TRUE<br>Output_Data = Set_Output<br>MSCY1M_Set_Output.req(Rem_Add:=0..125, Slot_Number:=ALL, Output_Data)<br>MSCY1M_Get_Input.req(Rem_Add:=0..125, Slot_Number:=ALL)<br>FSPMM1_SYNCH.ind (AREP) | COPERATE |
| 143 | COPERATE | DMPMM1_SYNCH.ind<br>/Go_AClr = True && CGC_Count = 0<br>=><br>Start Dx_Control_Interval_Timer (Bus_Para.Data_Control_Time/2)<br>UGC_Count:= UGC_Count+1<br>CGC_Count:= UGC_Count<br>DX_Lock:= TRUE<br>Output_Data = Set_Output<br>DMPMM1_Global_Control.req(Rem_Add:=127, Control_Command:=2,<br>Group_Select:=0)<br>MSCY1M_Set_Output.req(Rem_Add:=0..125, Slot_Number:=ALL, Output_Data)<br>MSCY1M_Get_Input.req(Rem_Add:=0..125, Slot_Number:=ALL)<br>FSPMM1_SYNCH.ind (AREP) | COPERATE |
| 144 | LDBP-OP-CPU | DMPMM1_Set_Bus_Par.cnf<br>=><br>FSPMM1_Load_Bus_Para.cnf(+) | OPERATE |
| 145 | DELSC-OP | DMPMM1_Delete_SC.cnf<br>=><br>FSPMM1_Delete_SC DP Master Cl1.cnf | OPERATE |
| 146 | RDVAL-OP | DMPMM1_Read_Value.cnf(Status,Value)<br>=><br>FSPMM1_Read_Value DP Master Cl1.cnf(Status, Value) | OPERATE |
| 147 | RESET | /spontanous<br>=><br>Act_Rem_Add:= 0<br>MSCY1M_Reset.req(Act_Rem_Add) | RESET-MS0 |
| 148 | RESET-MS0 | MSCY1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add:= Act_Rem_Add+1<br>MSCY1M_Reset.req(Act_Rem_Add) | RESET-MS0 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 149 | RESET-MS0 | MSCY1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MS1)=TRUE<br>=><br>Act_Rem_Add:= 0<br>MSAC1M_Reset.req(Act_Rem_Add) | RESET-MS1 |
| 150 | RESET-MS0 | MSCY1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MS1)=FALSE && &&<br>IsARExistent(MM1)=TRUE<br>=><br>MMAC1_Reset.req | RESET-MM1 |
| 151 | RESET-MS0 | MSCY1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add=125 && IsARExistent(MS1)=FALSE && &&<br>IsARExistent(MM1)=FALSE<br>=><br>DMPMM1_Reset.req | RESET-DMPM |
| 152 | RESET-MS1 | MSAC1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add=125<br>=><br>Act_Rem_Add:= 0<br>MSAL1M_Reset.req(Act_Rem_Add) | RESET-MS1 |
| 153 | RESET-MS1 | MSAL1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add<125<br>=><br>Act_Rem_Add:= Act_Rem_Add + 1<br>MSAL1M_Reset.req(Act_Rem_Add) | RESET-MS1 |
| 154 | RESET-MS1 | MSAL1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add = 125 && IsARExistent(MM1)=TRUE<br>=><br>MMAC1_Reset.req | RESET-MM1 |
| 155 | RESET-MS1 | MSAL1M_Reset.cnf(Act_Rem_Add)<br>/Act_Rem_Add = 125 && IsARExistent(MM1)=FALSE<br>=><br>DMPMM1_Reset.req | RESET-DMPM |
| 156 | RESET-MM1 | MMAC1_Reset.cnf<br>=><br>DMPMM1_Reset.req | RESET-DMPM |
| 157 | RESET-DMPM | DMPMM1_Reset.cnf<br>/USER_SetMode_Offline = TRUE \|\| USER_ChangedCritical_Para = TRUE | INIT-DMPM |
| 158 | RESET-DMPM | DMPMM1_Reset.cnf<br>/USER_SetMode_Offline = FALSE && USER_ChangedCritical_Para = FALSE<br>=><br>if(USER_sched_Reset = TRUE)<br>FSPMM1_Reset DP Master CI1.cnf<br>endif | POWER-ON |
| 159 | t state | FSPMM1_Abort DP Master CI1.req(AREP)<br>/AREP.AR_Type=MS0<br>=><br>Current Extended Function Num[CREP]:=No_Service<br>LR_State[CREP]:=W-LR-REQ<br>MSCY1M_Abort.req(Rem_Add:=CREP.Rem_Add) | SAME |
| 160 | t state | FSPMM1_Abort DP Master CI1.req(AREP)<br>/AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=No_Service<br>LR_State[CREP]:=W-LR-REQ<br>MSAC1M_Abort.req(Rem_Add:=CREP.Rem_Add) | SAME |
| 161 | t state | FSPMM1_Abort DP Master CI1.req(AREP)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Abort.req(Rem_Add:=CREP.Rem_Add) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 162 | t state | FSPMM1_Get_Slave_Diag.req(AREP, CREP)<br>/AREP.AR_Type=MS0<br>=><br>MSCY1M_Get_Slave_Diag.req(CREP.Rem_Add) | SAME |
| 163 | t state | MSCY1M_Get_Slave_Diag.cnf(+)(Rem_Add, Diag_Data)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Get_Slave_Diag.cnf(+)(AREP, CREP, Diag_Data) | SAME |
| 164 | t state | MSCY1M_Get_Slave_Diag.cnf(-)(Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Get_Slave_Diag.cnf(-)(AREP, CREP) | SAME |
| 165 | t state | MSCY1M_Set_Output.cnf(+)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = (Not Synchronized \|\| Enhanced Synchronized)<br>=><br>FSPMM1_Set_Output.cnf(+)(AREP, CREP, Slot_Number) | SAME |
| 166 | t state | MSCY1M_Set_Output.cnf(-)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = (Not Synchronized \|\| Enhanced Synchronized)<br>=><br>Status:= SC<br>FSPMM1_Set_Output.cnf(-)(AREP, CREP, Slot_Number, Status) | SAME |
| 167 | t state | MSCY1M_Get_Input.cnf(+)(Rem_Add, Slot_Number, Input_Data)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = (Not Synchronized \|\| Enhanced Synchronized)<br>=><br>FSPMM1_Get_Input.cnf(+)(AREP, CREP, Slot_Number, Input_Data) | SAME |
| 168 | t state | MSCY1M_Get_Input.cnf(-)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = (Not Synchronized \|\| Enhanced Synchronized)<br>=><br>Status:= SC<br>FSPMM1_Get_Input.cnf(-)(AREP, CREP, Slot_Number, Status) | SAME |
| 169 | t state | FSPMM1_Read.req(AREP, Slot_Number, Index, Length)<br>/AREP.AR_Type=MS1<br>=><br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length) | SAME |
| 170 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Read.cnf(+)(AREP, Length, Data) | SAME |
| 171 | t state | MSAC1M_Read.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Read.cnf(-)(AREP, Rem_Add, Error_Decode, Error_Code_1, Error_Code_2) | SAME |
| 172 | t state | FSPMM1_Write.req(AREP, Slot_Number, Index, Length, Data)<br>/AREP.AR_Type=MS1<br>=><br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data) | SAME |
| 173 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Write.cnf(+)(AREP, Length) | SAME |
| 174 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Write.cnf(-)(AREP, Rem_Add, Error_Decode, Error_Code_1, Error_Code_2) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 175 | t state | FSPMM1_Alarm_Ack.req(AREP, Slot_Number, Alarm_Type, Seq_Nr)<br>/AREP.AR_Type=MS1<br>=><br>MSAL1M_Alarm_Ack.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Alarm_Type, Seq_Nr) | SAME |
| 176 | t state | MSAL1M_Alarm_Ack.cnf(+)(Rem_Add, Slot_Number, Alarm_Type, Seq_Nr)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Alarm_Ack.cnf(+)(AREP, Slot_Number, Alarm_Type,Seq_Nr) | SAME |
| 177 | t state | MSAL1M_Alarm_Ack.cnf(-)(Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Alarm_Ack.cnf(-)(AREP, Slot Number, Alarm Type,Seq Nr) | SAME |
| 178 | t state | FSPMM1_Get_Master_Diag.rsp(+)(AREP, Diagnosis_Data)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Get_Master_Diag.rsp(Status:=OK, Diagnosis_Data) | SAME |
| 179 | t state | FSPMM1_Get_Master_Diag.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Get_Master_Diag.rsp(Status) | SAME |
| 180 | t state | MMAC1_Get_Master_Diag.ind(Identifier)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Get_Master_Diag.ind(AREP, Mdiag_Identifier:=Identifier) | SAME |
| 181 | t state | FSPMM1_Start_Seq.rsp(+)(AREP, Max_Len_Data_Unit)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Start_Seq.rsp(Status:=OK, Max_Length_Data_Unit) | SAME |
| 182 | t state | FSPMM1_Start_Seq.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Start_Seq.rsp(Status) | SAME |
| 183 | t state | MMAC1_Start_Seq.ind(Req_Add, Area_Code, Timeout)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Start_Seq.ind(AREP, Area_Code, Timeout) | SAME |
| 184 | t state | FSPMM1_Download.rsp(+)(AREP)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Download.rsp(Status:=OK) | SAME |
| 185 | t state | FSPMM1_Download.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Download.rsp(Status) | SAME |
| 186 | t state | MMAC1_Download.ind(Req_Add, Area_Code, Address_Offset, Data)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Download.ind(AREP, Area_Code, Add_Offset, Data) | SAME |
| 187 | t state | FSPMM1_Upload.rsp(+)(AREP, Data)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Upload.rsp(Status:=OK, Data) | SAME |
| 188 | t state | FSPMM1_Upload.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Upload.rsp(Status) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 189 | t state | MMAC1_Upload.ind(Req_Add, Area_Code, Address_Offset, Data_Length)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Upload.ind(AREP, Area_Code, Add_Offset, Data_Len:=Data_Length) | SAME |
| 190 | t state | FSPMM1_End_Seq.rsp(+)(AREP)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_End_Seq.rsp(Status:=OK) | SAME |
| 191 | t state | FSPMM1_End_Seq.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_End_Seq.rsp(Status) | SAME |
| 192 | t state | MMAC1_End_Seq.ind(Req_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_End_Seq.ind(AREP) | SAME |
| 193 | t state | FSPMM1_Act_Param.rsp(+)(AREP)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Act_Param.rsp(Status:=OK) | SAME |
| 194 | t state | FSPMM1_Act_Param.rsp(-)(AREP, Status)<br>/AREP.AR_Type=MM1<br>=><br>MMAC1_Act_Param.rsp(Status) | SAME |
| 195 | t state | MMAC1_Act_Param.ind(Area_Code, Activate)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Act_Param.ind(AREP, Area_Code, Activate) | SAME |
| 196 | t state | MMAC1_Act_Para_Brct.ind(Area_Code)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Act_Para_Brct.ind(AREP, Area_Code) | SAME |
| 197 | t state | MSAC1M_Reject.ind(Rem_Add, Status)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_DP Master Cl1 Reject.ind(AREP, Reason:=Status) | SAME |
| 198 | t state | MSAL1M_Started.ind(Rem_Add, Alarm_Limit)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_DP Master Cl1 Started.ind(AREP, Alarm_Limit) | SAME |
| 199 | t state | MSAL1M_Stopped.ind(Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>Current Extended Function Num[CREP]:=No_Service<br>LR_State[CREP]:=W-LR-REQ<br>FSPMM1_DP Master Cl1 Stopped.ind(AREP) | SAME |
| 200 | t state | MSAL1M_Alarm_Notification.ind(Rem_Add, Alarm_Type, Slot_Number, Seq_Nr, Alarm_Specifier, Add_Ack, Alarm_Data)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_Alarm_Notification.ind(AREP, Alarm_Type, Slot_Number, Seq_Nr, Alarm_Specifier, Add_Ack, Alarm_Data) | SAME |
| 201 | t state | MSCY1M_New_Slave_Diag.ind(Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_New_Slave_Diag.ind(AREP, CREP) | SAME |
| 202 | t state | MSCY1M_New_Input.ind(Rem_Add)<br>/SetContext(Rem_Add, Service)=TRUE<br>=><br>FSPMM1_New_Input.ind(AREP, CREP) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 203 | t state | FSPMM1_Set_Output.req(AREP, CREP, Slot_Number, Output_Data, Final)<br>/AREP.AR_Type=MS0 &&<br>(DX_Lock \|\| Chg_Buffer)<br>=><br>Status:= SE<br>FSPMM1_Set_Output.cnf(-)(AREP, CREP, Slot_Number, Status) | SAME |
| 204 | t state | FSPMM1_Get_Input.req(AREP, CREP, Slot_Number)<br>/AREP.AR_Type=MS0 &&<br>(DX_Lock \|\| Chg_Buffer)<br>=><br>Status:= SE<br>FSPMM1_Get_Input.cnf(-)(AREP, CREP, Slot_Number, Status) | SAME |
| 205 | t state | FSPMM1_Set_Output.req(AREP, CREP, Slot_Number, Output_Data, Final)<br>/AREP.AR_Type=MS0 &&<br>!DX_Lock && !Chg_Buffer && Isochronous Mode <> Buffered Synchronized<br>=><br>MSCY1M_Set_Output.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Output_Data) | SAME |
| 206 | t state | FSPMM1_Get_Input.req(AREP, CREP, Slot_Number)<br>/AREP.AR_Type=MS0 &&<br>!DX_Lock && !Chg_Buffer && Isochronous Mode <> Buffered Synchronized<br>=><br>MSCY1M_Get_Input.req(Rem_Add:=CREP.Rem_Add, Slot_Number) | SAME |
| 207 | t state | FSPMM1_Set_Output.req(AREP, CREP, Slot_Number, Output_Data, Final)<br>/AREP.AR_Type=MS0 &&<br>!DX_Lock && !Chg_Buffer && Isochronous Mode = Buffered Synchronized<br>=><br>Bfr_Set_Output:= Output_Data<br>Chg_Buffer:= Final<br>FSPMM1_Set_Output.cnf(+)(AREP, CREP, Slot_Number) | SAME |
| 208 | t state | FSPMM1_Get_Input.req(AREP, CREP, Slot_Number)<br>/AREP.AR_Type=MS0 &&<br>!DX_Lock && !Chg_Buffer && Isochronous Mode = Buffered Synchronized<br>=><br>Input_Data:= Bfr_Get_Input<br>FSPMM1_Get_Input.cnf(+)(AREP, CREP, Slot_Number, Input_Data) | SAME |
| 209 | t state | MSCY1M_Set_Output.cnf(+)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized<br>=> | SAME |
| 210 | t state | MSCY1M_Set_Output.cnf(-)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Set_Output = Nil | SAME |
| 211 | t state | MSCY1M_Get_Input.cnf(+)(Rem_Add, Slot_Number, Input_Data)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized &&<br>NOT received all MSCY1M_Get_Input.cnf<br>=><br>Get_Input = Input_Data | SAME |
| 212 | t state | MSCY1M_Get_Input.cnf(+)(Rem_Add, Slot_Number, Input_Data)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized&&<br>NOT received all MSCY1M_Get_Input.cnf<br>=><br>Chg_Buffer:= FALSE<br>Get_Input = Input_Data | SAME |
| 213 | t state | MSCY1M_Get_Input.cnf(-)(Rem_Add, Slot_Number)<br>/SetContext(Rem_Add, Service)=TRUE &&<br>Isochronous Mode = Buffered Synchronized<br>=><br>Get_Input = Nil | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 214 | ANY-STATE | DMPMM1_Event.ind(Event, Add_Info)<br>=><br>FSPMM1_DP Master Cl1 Event.ind(Event, Add_Info) | SAME |
| 215 | ANY-STATE | DMPMM1_SYNCH_Delayed.ind (TSH)<br>=><br>FSPMM1_SYNCH_Delayed.ind (AREP, TSH) | SAME |
| 216 | ANY-STATE | DMPMM1_Fault.ind<br>=><br>USER_sched_Reset:= False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 217 | ANY-STATE | MSCY1M_Fault.ind<br>=><br>USER_sched_Reset:= False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 218 | ANY-STATE | MSAL1M_Fault.ind<br>=><br>USER_sched_Reset:= False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 219 | ANY-STATE | MSAC1M_Fault.ind<br>=><br>USER_sched_Reset:= False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 220 | ANY-STATE | MMAC1_Fault.ind<br>=><br>USER_sched_Reset:= False<br>FSPMM1_DP Master Cl1 Fault.ind | RESET |
| 221 | ANY-STATE | FSPMM1_Reset DP Master Cl1.req<br>=><br>USER_sched_Reset:= True | RESET |
| 222 | ANY-STATE | DMPMM1_SYCL_Time_Event.cnf (Send Delay Time, Status)<br>=><br>FSPMM1_SYCL_Time_Event.cnf (AREP, Send Delay Time, Status) | SAME |
| 223 | ANY-STATE | FSPMM1_SYCL_Time_Event.req (AREP)<br>=><br>DMPMM1_SYCL_Time_Event.req | SAME |
| 224 | ANY-STATE | FSPMM1_SYCL_Clock_Value.req (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status)<br>=><br>DMPMM1_SYCL_Clock_Value.req (Clock Value Time Event, Clock Value previous TE, Clock Value Status) | SAME |
| 225 | ANY-STATE | DMPMM1_SYCL_Clock_Value.cnf (Status)<br>=><br>FSPMM1_SYCL_Clock_Value.cnf (AREP, Status) | SAME |
| 226 | ANY-STATE | DMPMM1_SYCL_Clock_Value.ind (Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg)<br>=><br>FSPMM1_SYCL_Clock_Value.ind (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg) | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 227 | t state | FSPMM1_Initiate_Load.req ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Initiate_Load<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=10+sizeof(User Specific)<br>Data.w.IL.Extended_Function_Num:=Initiate_Load<br>Data.w.IL.LR_Index:=LR Index<br>Data.w.IL.Load_Type:=Load Type<br>Data.w.IL.Load_Image_Size:=Load Image Size<br>Data.w.IL.User_Specific:=User Specific<br>Data.w.IL.Intersegment_Request_Timeout:=Intersegment Request Timeout<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 228 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Initiate_Load<br>=><br>Slot_Number:=Current Slot Number[CREP]<br>Index:=255<br>Length:=10<br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 229 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Initiate_Load<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Actual LR Size:=Data.r.IL.Actual_LR_Size<br>Max Response Delay:=Data.r.IL.Max_Response_Delay<br>Max Segment Length:=Data.r.IL.Max_Segment_Length<br>User Specific:=Data.r.IL.User_Specific<br>FSPMM1_Initiate_Load.cnf(+) ( AREP, Actual LR Size, Max Response Delay, Max Segment Length, User Specific )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 230 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Initiate_Load<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Initiate_Load.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 231 | t state | FSPMM1_Pull_Segment.req ( AREP, Slot Number, LR Index, Segment Length )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Pull<br>Current LR Index[CREP]:=LR Index<br>Index:=255<br>Length:=Segment Length<br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length)<br>LR_State[CREP]:=W-LR-CNF | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 232 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Pull<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Segment Number:=Data.PULL.Sequence Number<br>More Follows:=Data.PULL.Options.More Follows<br>Data:=Data.PULL.Region Data<br>Segment Length:=Length-6<br>FSPMM1_Pull_Segment.cnf(+) ( AREP, Segment Length, Segment Number, More Follows, Data )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 233 | t state | MSAC1M_Read.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Pull<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Pull_Segment.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 234 | t state | FSPMM1_Push_Segment.req ( AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Push<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=6+Segment Length<br>Data.PUSH.Extended_Function_Num:=Push<br>Data.PUSH.LR_Index:=LR Index<br>Data.PUSH.Segment_Number:=Segment Number<br>Data.PUSH.Option.More_Follows:=More Follows<br>Data.PUSH.Data:=Data<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 235 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Push<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Segment Number:=Data.PUSH.Sequence Number<br>More Follows:=Data.PUSH.Options.More Follows<br>Data:=Data.PUSH.Region Data<br>Segment Length:=Length-6<br>FSPMM1_Push_Segment.cnf(+) ( AREP)<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 236 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Push<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Push_Segment.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 237 | t state | FSPMM1_Terminate_Load.req ( AREP, Slot Number, LR Index )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Terminate_Load<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=6<br>Data.TL.Extended_Function_Num:=Terminate_Load<br>Data.TL.LR_Index:=LR Index<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 238 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Terminate_Load<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Terminate_Load.cnf(+) ( AREP)<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 239 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Terminate_Load<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Terminate_Load.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 240 | t state | FSPMM1_Call.req ( AREP, Slot Number, Entity Number, FI Index, Execution Argument)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Call<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.w.CALL.Extended_Function_Num:=Call<br>Data.w.CALL.Entity_Number:= Entity_Number<br>Data.w.CALL.FI_Index:=FI Index<br>Data.w.CALL.Execution_Argument:=Execution Argument<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 241 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Call<br>=><br>Slot_Number:=Current Slot Number[CREP]<br>Index:=255<br>Length:=CREP.Max_PDU_Length<br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 242 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Call<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Result Argument:=Data.r.CALL.Result_Argument<br>FSPMM1_Call.cnf(+) ( AREP, Result Argument )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 243 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Call<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Call.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 244 | t state | FSPMM1_Start.req ( AREP, Slot Number, FI Index, Execution Argument)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Start<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.FIS.Extended_Function_Num:=Start<br>Data.FIS.FI_Index:=FI Index<br>Data.FIS.Execution_Argument:=Execution Argument<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 245 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Start<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Start.cnf(+) ( AREP )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 246 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Start<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Start.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 247 | t state | FSPMM1_Stop.req ( AREP, Slot Number, FI Index)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Stop<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.FIS.Extended_Function_Num:=Stop<br>Data.FIS.FI_Index:=FI Index<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 248 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Stop<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Stop.cnf(+) ( AREP )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 249 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Stop<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Stop.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 250 | t state | FSPMM1_Resume.req ( AREP, Slot Number, FI Index, Execution Argument)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Resume<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.FIS.Extended_Function_Num:=Resume<br>Data.FIS.FI_Index:=FI Index<br>Data.FIS.Execution_Argument:=Execution Argument<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 251 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Resume<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Resume.cnf(+) ( AREP )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 252 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Resume<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Resume.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 253 | t state | FSPMM1_Reset.req ( AREP, Slot Number, FI Index)<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Reset<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.FIS.Extended_Function_Num:=Reset<br>Data.FIS.FI_Index:=FI Index<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 254 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Reset<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FSPMM1_Reset.cnf(+) ( AREP )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 255 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Reset<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Reset.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 256 | t state | FSPMM1_Get_FI_State.req ( AREP, Slot Number, FI Index )<br>/LR_State[CREP]==W-LR-REQ<br>&AREP.AR_Type=MS1<br>=><br>Current Extended Function Num[CREP]:=Get_State<br>Current Slot Number[CREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.w.STATE.Extended_Function_Num:=Get_State<br>Data.w.STATE.FI_Index:=FI Index<br>MSAC1M_Write.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length, Data)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 257 | t state | MSAC1M_Write.cnf(+)(Rem_Add, Length)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Get_State<br>=><br>Slot_Number:=Current Slot Number[CREP]<br>Index:=255<br>Length:=5<br>MSAC1M_Read.req(Rem_Add:=CREP.Rem_Add, Slot_Number, Index, Length)<br>LR_State[CREP]:=W-LR-CNF | SAME |
| 258 | t state | MSAC1M_Read.cnf(+)(Rem_Add, Length, Data)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Get_State<br>=><br>Current Extended Function Num[CREP]=No_Service<br>FI State:=Data.r.STATE.State<br>FSPMM1_Get_FI_State.cnf (+) ( AREP, FI State )<br>LR_State[CREP]:=W-LR-REQ | SAME |
| 259 | t state | MSAC1M_Write.cnf(-)(Rem_Add, Error_Decode, Error_Code_1, Error_Code_2)<br>/LR_State[CREP]==W-LR-CNF<br>&SetContext(Rem_Add, Service)=TRUE &&<br>Current Extended Function Num[CREP]=Get_State<br>=><br>Current Extended Function Num[CREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM1_Get_State.cnf(-) ( AREP, Error Code )<br>LR_State[CREP]:=W-LR-REQ | SAME |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 8.2.4 Fonctions

Le Tableau 40 contient les fonctions utilisées par le FSPMM1, leurs arguments et leurs descriptions.

**Tableau 40 – Fonctions utilisées par le FSPMM1**

| Nom de fonction | Description |
|---|---|
| SetContext(Rem_Add, Service) | Cette fonction vérifie s'il existe une entrée saisie pour les entrées Rem_Add respectivement Res_SAP et le Service dans l'ARL et la CRL correspondante du maître DP (Classe 1).<br>A) S'il existe une entrée, elle retourne TRUE et règle contexte local selon le CREP et l'AREP courants.<br>B) Autrement, elle retourne FALSE. |
| IsARexistent (AR) | Cette fonction vérifie s'il existe une entrée avec ARtype=AR. |
| Chg_Bfr(Buffer1, Buffer 2) | Cette fonction échange Buffer 1 et Buffer 2. |

## 8.3 FSPMM2

### 8.3.1 Définitions des primitives

#### 8.3.1.1 Primitives échangées entre AP-Context et FSPMM2

Le Tableau 41 montre les primitives de service, y compris leurs paramètres associés, émises par l'AP-Context et reçues par le FSPMM2.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 41 – Primitives émises par l'AP-Context vers le FSPMM2**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Init DP master CI2.req | AP-Context | Bus Para | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3 et dans la CEI 61158-3-3 |
| Reset DP master CI2.req | AP-Context | (aucun) | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3. |
| Abort.req | AP-Context | AREP<br>Subnet,<br>Instance,<br>Reason Code | |
| Read Slave Diag.req | AP-Context | AREP | |
| Read Output.req | AP-Context | AREP | |
| Read Input.req | AP-Context | AREP | |
| Get Cfg.req | AP-Context | AREP | |
| Set Slave Add.req | AP-Context | AREP,<br>New Slave Add,<br>Ident Number,<br>No Add Chg,<br>Rem Slave Data | |
| Initiate.req | AP-Context | AREP,<br>Send Timeout,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Read.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length | |
| Write.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Data Transport.req | AP-Context | AREP,<br>Slot Number,<br>Index,<br>Length,<br>Data | |
| Get Master Diag.req | AP-Context | AREP,<br>MDiag Identifier | |
| Start Seq.req | AP-Context | AREP,<br>Area Code,<br>Timeout | |
| Download.req | AP-Context | AREP,<br>Area Code,<br>Add Offset,<br>Data | |
| Upload.req | AP-Context | AREP,<br>Area Code,<br>Add Offset,<br>Data Len | |
| End Seq.req | AP-Context | AREP | |
| Act Param.req | AP-Context | AREP,<br>Area Code,<br>Activate | |
| Act Para Brct.req | AP-Context | AREP,<br>Area Code | |
| Load ARL DP master CI2.req | AP-Context | List of ARL Entries | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Get ARL DP master CI2.req | AP-Context | (aucun) | |
| Load CRL DP master CI2.req | AP-Context | List of CRL Entries | |
| Get CRL DP master CI2.req | AP-Context | (aucun) | |
| Initiate Load.req | AP-Context | AREP,<br>Slot Number,<br>LR Index,<br>Load Type,<br>Load Image Size,<br>Intersegment Request,<br>Timeout | |
| Pull Segment.req | AP-Context | AREP,<br>Slot Number,<br>LR Index,<br>Segment Length | |
| Push Segment.req | AP-Context | AREP,<br>Slot Number<br>LR Index,<br>Segment Length<br>Segment Number,<br>More Follows,<br>Data | |
| Terminate Load.req | AP-Context | AREP<br>Slot Number<br>LR Index | |
| Start.req | AP-Context | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Stop.req | AP-Context | AREP<br>Slot Number<br>FI Index | |
| Resume.req | AP-Context | AREP<br>Slot Number<br>FI Index<br>Execution Argument | |
| Reset.req | AP-Context | AREP<br>Slot Number<br>FI Index | |
| Call.req | FSPMS | AREP<br>Slot Number<br>Entity Number<br>FI Index<br>Execution Argument | |

Le Tableau 42 montre les primitives de service, y compris leurs paramètres associés, émises par l'AP-Context et reçues par le FSPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Tableau 42 – Primitives émises par le FSPMM2 vers l'AP-Context**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Init DP master CI2.cnf | FSPMM2 | (aucun) | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3. |
| Reset.cnf | FSPMM2 | (aucun) | |
| Read Slave Diag.cnf(+) | FSPMM2 | AREP,<br>Diag Data | |
| Read Slave Diag.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Get Cfg.cnf(+) | FSPMM2 | AREP,<br>Cfg Data | |
| Get Cfg.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Read Output.cnf(+) | FSPMM2 | AREP,<br>Output Data | |
| Read Output.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Read Input.cnf(+) | FSPMM2 | AREP,<br>Input Data | |
| Read Input.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Set Slave Add.cnf(+) | FSPMM2 | AREP | |
| Set Slave Add.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Initiate.cnf(+) | FSPMM2 | AREP,<br>Max Len Data Unit,<br>Features Supported,<br>Profile Features Supported,<br>Profile Ident Number,<br>Add Addr Param | |
| Initiate.cnf(-) | FSPMM2 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Read.cnf(+) | FSPMM2 | AREP,<br>Length,<br>Data | |
| Read.cnf(-) | FSPMM2 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.cnf(+) | FSPMM2 | AREP,<br>Length | |
| Write.cnf(-) | FSPMM2 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Data Transport.cnf(+) | FSPMM2 | AREP,<br>Length,<br>Data | |
| Data Transport.cnf(-) | FSPMM2 | AREP,<br>Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Get Master Diag.cnf(+) | FSPMM2 | AREP,<br>Diagnosis Data | |
| Get Master Diag.cnf(-) | FSPMM2 | AREP,<br>Status | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Start Seq.cnf(+) | FSPMM2 | AREP,<br>Max Len Data Unit | |
| Start Seq.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Download.cnf(+) | FSPMM2 | AREP | |
| Download.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Upload.cnf(+) | FSPMM2 | AREP,<br>Data | |
| Upload.cnf(-) | FSPMM2 | AREP,<br>Status | |
| End Seq.cnf(+) | FSPMM2 | AREP | |
| End Seq.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Act Param.cnf(+) | FSPMM2 | AREP | |
| Act Param.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Act Para Brct.cnf(+) | FSPMM2 | AREP | |
| Act Para Brct.cnf(-) | FSPMM2 | AREP,<br>Status | |
| Event.ind | FSPMM2 | Event,<br>Add Info | |
| DP master Cl2 Reject.ind | FSPMM2 | AREP,<br>Reason | |
| Abort.ind | FSPMM2 | AREP,<br>Locally Generated,<br>Subnet,<br>Instance,<br>Reason_Code,<br>Additional_Detail | |
| DP master Cl2 Fault.ind | FSPMM2 | (aucun) | |
| Load ARL DP master Cl2.cnf(+) | FSPMM2 | (aucun) | |
| Load ARL DP master Cl2.cnf(-) | FSPMM2 | Status | |
| Get ARL DP master Cl2.cnf(+) | FSPMM2 | List of ARL Entries | |
| Get ARL DP master Cl2.cnf(-) | FSPMM2 | Status | |
| Load CRL DP master Cl2.cnf(+) | FSPMM2 | (aucun) | |
| Load CRL DP master Cl2.cnf(-) | FSPMM2 | Status | |
| Get CRL DP master Cl2.cnf(+) | FSPMM2 | List of CRL Entries | |
| Get CRL DP master Cl2.cnf(-) | FSPMM2 | Status | |
| Initiate Load.cnf(+) | FSPMM2 | AREP,<br>Actual LR Size,<br>Max Response Delay,<br>Max Segment Length | |
| Initiate Load.cnf(-) | FSPMM2 | AREP,<br>Error Code | |
| Pull Segment.cnf(+) | FSPMM2 | AREP,<br>Segment Length,<br>Segment Number,<br>More Follows,<br>Data | |
| Pull Segment.cnf(-) | FSPMM2 | AREP,<br>Error Code | |
| Push Segment.cnf(+) | FSPMM2 | AREP | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Push Segment.cnf(-) | FSPMM2 | AREP,<br>Error Code | |
| Terminate Load.cnf(+) | FSPMM2 | AREP | |
| Terminate Load.cnf(-) | FSPMM2 | AREP,<br>Error Code | |
| Start.cnf(+) | FSPMM2 | AREP | |
| Start.cnf(-) | FSPMM2 | AREP<br>Error Code<br>Function Invocation State | |
| Stop.cnf(+) | FSPMM2 | AREP | |
| Stop.cnf(-) | FSPMM2 | AREP<br>Error Code<br>Function Invocation State | |
| Resume.cnf(+) | FSPMM2 | AREP | |
| Resume.cnf(-) | FSPMM2 | AREP<br>Error Code<br>Function Invocation State | |
| Reset.cnf(+) | FSPMM2 | AREP | |
| Reset.cnf(-) | FSPMM2 | AREP<br>Error Code | |
| Call.cnf(+) | FSPMM2 | AREP<br>Result Argument | |
| Call.cnf(-) | FSPMM2 | AREP<br>Result Argument<br>Error Code | |

#### 8.3.1.2 Paramètres des primitives de FSPMM2

Les paramètres utilisés avec les primitives échangées entre le FSPMM2 et l'AP-Context sont décrits dans "Définition des services de la FAL" (voir la CEI 61158-5-3).

### 8.3.2 Description de diagramme d'états

Ce Diagramme est utilisé pour coordonner les interactions entre l'AP-Context (Contexte d'AP) et les DMPMM1, MSCY1M, MSAC1M, MSAL1M et MMAC1. Sachant que la prise en charge de plusieurs diagrammes d'états pour la communication avec un esclave pris séparément ainsi que les contraintes de synchronisation du fonctionnement de l'esclave exigent une programmation cohérente, cette tâche est accomplie par le FSPMM1 également.

**Variables locales**

**LR_State**
(Array of Unsigned8)

Cette variable stocke l'état de la séquence Load Region de l'AREP correspondant.

**CurrentExtendedFunctionNum**
(Array of Unsigned 8)

Cette variable stocke le numéro de fonction étendu du service LR actuellement traité de l'AREP correspondant.

**CurrentSlotNumber**
(Array of Unsigned8)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Cette variable stocke le numéro de position du service LR actuellement traité de l'AREP correspondant.

### 8.3.3 Table d'états de FSPMM2

Le Tableau 43 contient la description complète du diagramme d'états FSPMM2.

**Tableau 43 – Table d'états de FSPMM2**

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 1 | POWER-ON | Load ARL DP master CI2.req (List of ARL Entries)<br>/paramètres valides<br>=><br>Load ARL DP master CI2.cnf(+) | POWER-ON |
| 2 | POWER-ON | Load ARL DP master CI2.req (List of ARL Entries)<br>/paramètres non valides<br>=><br>Status:= NO<br>Load ARL DP master CI2.cnf(-) (Status) | POWER-ON |
| 3 | POWER-ON | Get ARL DP master CI2.req<br>/ARL loaded<br>=><br>Get ARL DP master CI2.cnf(+) ( List of ARL Entries ) | POWER-ON |
| 4 | POWER-ON | Get ARL DP master CI2.req<br>/ARL not loaded<br>=><br>Status:= NO<br>Get ARL DP master CI2.cnf(-) (Status) | POWER-ON |
| 5 | POWER-ON | Load CRL DP master CI2.req ( List of CRL Entries )<br>/paramètres valides<br>=><br>Load CRL DP master CI2.cnf(+) | POWER-ON |
| 6 | POWER-ON | Load CRL DP master CI2.req ( List of CRL Entries )<br>/paramètres non valides<br>=><br>Status:= NO<br>Load CRL DP master CI2.cnf(-) (Status) | POWER-ON |
| 7 | POWER-ON | Get CRL DP master CI2.req<br>/CRL loaded<br>=><br>Get CRL DP master CI2.cnf(+) (List of CRL Entries) | POWER-ON |
| 8 | POWER-ON | Get CRL DP master CI2.req<br>/CRL not loaded<br>=><br>Status:= NO<br>Get CRL DP master CI2.cnf(-) (Status) | POWER-ON |
| 9 | PON | FSPMM2_Init DP master CI2 .req(Bus_Para)<br>=><br>StoreNumberIssuedMInit()<br>Current Extended Function Num[AREP]=No_Service<br>LR_State[AREP]:=W-LR-REQ<br>DMPMM2_Minit_DLL.req(Bus_Para)<br>MMAC_MInit_MM.req<br>MSAC2M_MInit_MS2.req(First_MS2_AREP ..Last_MS2_AREP) | INIT-DLM |
| 10 | INIT-DLM | DMPMM2_Minit_DLL.cnf<br>=><br>ignore | INIT-DLM |
| 11 | INIT-DLM | MMAC2_MInit_MM.cnf<br>=><br>ignore | INIT-DLM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 12 | INIT-DLM | MSAC2M_MInit_MS2.cnf (CREP)<br>/IsLastWaitingMInit()=FALSE<br>=><br>ignore | INIT-DLM |
| 13 | INIT-DLM | MSAC2M_MInit_MS2.cnf (CREP)<br>/IsLastWaitingMInit()=TRUE<br>=><br>FSPMM2_Init DP master CI2 .cnf | RUN |
| 14 | RUN | Any FSPMM2.req<br>/wrong AREP.AR_Type<br>=><br>ignore | RUN |
| 15 | RUN | Any MMAC1.cnf<br>/SetContext(Rem_add, Service) = FALSE<br>=><br>ignore | RUN |
| 16 | RUN | Any MSAC2M.req<br>/SetContext(CREP, Service) = FALSE<br>=><br>ignore | RUN |
| 17 | RUN | Any DMPMM2.req<br>/SetContext(Rem_add, Service) = FALSE<br>=><br>ignore | RUN |
| 18 | RUN | FSPMM2_Abort.req(AREP, Subnet, Instance, Reason_Code)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Abort.req | RUN |
| 19 | RUN | FSPMM2_Abort.req(AREP, Subnet, Instance, Reason_Code)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Abort.req | RUN |
| 20 | RUN | FSPMM2_Abort.req(AREP, Subnet, Instance, Reason_Code)<br>/AREP.AR_Type = MS2<br>=><br>Current Extended Function Num[AREP]=No_Service<br>LR_State[AREP]:=W-LR-REQ<br>MSAC2M_Abort.req(CREP, Subnet, Instance, Reason_Code) | RUN |
| 21 | RUN | MSAC2M_Abort.ind(CREP, Locally_Generated, Subnet, Instance, Reason_Code, Abort_Detail)<br>/SetContext(CREP, Service) = TRUE<br>=><br>Current Extended Function Num[AREP]=No_Service<br>LR_State[AREP]:=W-LR-REQ<br>FSPMM2_Abort.ind(AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) | RUN |
| 22 | RUN | DMPMM2_Event.ind(Event, Add_info)<br>=><br>FSPMM2_Event.ind(Event=Event/Fault, Add_Info=Add_info) | RUN |
| 23 | RUN | FSPMM2_Read_Slave_Diag.req(AREP)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Read_Slave_Diag.req(CREP.Rem_Add) | RUN |
| 24 | RUN | DMPMM2_Read_Slave_Diag.cnf(+)(Rem_Add, Diag_Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Slave_Diag.cnf(+)(AREP, Diag_Data) | RUN |
| 25 | RUN | DMPMM2_Read_Slave_Diag.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Slave_Diag.cnf(-)(AREP, Status=Status) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 26 | RUN | FSPMM2_Get_Cfg.req(AREP)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Get_Cfg.req(CREP.Rem_Add) | RUN |
| 27 | RUN | DMPMM2_Get_Cfg.cnf(+)(Rem_Add, Cfg_Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Get_Cfg.cnf(+)(AREP, Cfg_Data) | RUN |
| 28 | RUN | DMPMM2_Get_Cfg.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Get_Cfg.cnf(-)(AREP, Status) | RUN |
| 29 | RUN | FSPMM2_Read_Input.req(AREP)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Read_Input.req(CREP.Rem_Add) | RUN |
| 30 | RUN | DMPMM2_Read_Input.cnf(+)(Rem_Add, Inp_Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Input.cnf(+)(AREP, Inp_Data) | RUN |
| 31 | RUN | DMPMM2_Read_Input.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Input.cnf(-)(AREP, Status) | RUN |
| 32 | RUN | FSPMM2_Read_Output.req(AREP)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Read_Output.req(CREP.Rem_Add) | RUN |
| 33 | RUN | DMPMM2_Read_Output.cnf(+)(Rem_Add, Outp_Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Output.cnf(+)(AREP, Outp_Data) | RUN |
| 34 | RUN | DMPMM2_Read_Output.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Read_Output.cnf(-)(AREP, Status) | RUN |
| 35 | RUN | FSPMM2_Set_Slave_Add.req(AREP, New_Slave_Add, Ident_Number, No_Add_Chg, Rem_Slave_Data)<br>/AREP.AR_Type = MS0<br>=><br>DMPMM2_Set_Slave_Add.req(CREP.Rem_Add, New_Slave_Add, Ident_Number, No_Add_Chg, Rem_Slave_Data) | RUN |
| 36 | RUN | DMPMM2_Set_Slave_Add.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Set_Slave_Add.cnf(+)(AREP) | RUN |
| 37 | RUN | DMPMM2_Set_Slave_Add.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Set_Slave_Add.cnf(-)(AREP, Status=Status) | RUN |
| 38 | RUN | Any DMPMM2.req<br>/SetContext(Rem_add, Service) = FALSE<br>=><br>ignore | RUN |
| 39 | RUN | FSPMM2_Initiate.req(AREP, Send Timeout, Features Supported, Profile Features Supported,<br>Profile Ident Number, Add Addr Param)<br>/AREP.AR_Type = MS2<br>=><br>MSAC2M_Initiate.req(CREP, Send Timeout, Features Supported, Profile Features Supported,<br>Profile Ident Number, Add Addr Param) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 40 | RUN | MSAC2M_Initiate.cnf(+)(CREP, Max Len Data Unit, Features Supported, Profile Features Supported,<br>Profile Ident Number, Add Addr Param)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Initiate.cnf(+)(AREP, Max Len Data Unit, Features Supported, Profile Features Supported,<br>Profile Ident Number, Add Addr Param) | RUN |
| 41 | RUN | MSAC2M_Initiate.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Initiate.cnf(-)(AREP, Error Decode, Error Code 1, Error Code 2) | RUN |
| 42 | RUN | FSPMM2_Read.req(AREP, Slot Number, Index, Length)<br>/AREP.AR_Type = MS2<br>=><br>MSAC2M_Read.req(CREP, Slot Number, Index, Length) | RUN |
| 43 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Read.cnf(+)(AREP, Length, Data) | RUN |
| 44 | RUN | MSAC2M_Read.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Read.cnf(-)(AREP, Error Decode, Error Code 1, Error Code 2) | RUN |
| 45 | RUN | FSPMM2_Write.req(AREP, Slot Number, Index, Length, Data)<br>/AREP.AR_Type = MS2<br>=><br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data) | RUN |
| 46 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Write.cnf(+)(AREP, Length) | RUN |
| 47 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Write.cnf(-)(AREP, Error Decode, Error Code 1, Error Code 2) | RUN |
| 48 | RUN | FSPMM2_Data_Transport.req(AREP, Slot Number, Index, Length, Data)<br>/AREP.AR_Type = MS2<br>=><br>MSAC2M_Data_Transport.req(CREP, Slot Number, Index, Length, Data) | RUN |
| 49 | RUN | MSAC2M_Data_Transport.cnf(+)(CREP, Length, Data)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Data_Transport.cnf(+)(AREP, Length, Data) | RUN |
| 50 | RUN | MSAC2M_Data_Transport.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2_Data_Transport.cnf(-)(AREP, Error Decode, Error Code 1, Error Code 2) | RUN |
| 51 | RUN | MSAC2M_Closed.ind(CREP)<br>/SetContext(CREP, Service) = TRUE<br>=><br>FSPMM2 DP master CI2 _Closed.ind(AREP) | RUN |
| 52 | RUN | MMAC2_Reject.ind(Rem_Add, Reason_Code)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2 DP master CI2 _Reject.ind(AREP, Reason) | RUN |
| 53 | RUN | FSPMM2_Get_Master_Diag.req(AREP, MDiag Identifier)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Get_Master_Diag.req(AREP, MDiag Identifier) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 54 | RUN | MMAC2_Get_Master_Diag.cnf(+)(Rem_Add, Diagnosis Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Get_Master_Diag.cnf(+)(AREP, Diagnosis Data) | RUN |
| 55 | RUN | MMAC2_Get_Master_Diag.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Get_Master_Diag.cnf(-)(AREP, Status) | RUN |
| 56 | RUN | FSPMM2_Start_Seq.req(AREP, Area Code, Timeout)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Start_Seq.req(AREP, Area Code, Timeout) | RUN |
| 57 | RUN | MMAC2_Start_Seq.cnf(+)(Rem_Add, Max Len Data Unit)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Start_Seq.cnf(+)(AREP, Max Len Data Unit) | RUN |
| 58 | RUN | MMAC2_Start_Seq.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Start_Seq.cnf(-)(AREP, Status) | RUN |
| 59 | RUN | FSPMM2_End_Seq.req(AREP)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_End_Seq.req(AREP) | RUN |
| 60 | RUN | MMAC2_End_Seq.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_End_Seq.cnf(+)(AREP) | RUN |
| 61 | RUN | MMAC2_End_Seq.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_End_Seq.cnf(-)(AREP, Status) | RUN |
| 62 | RUN | FSPMM2_Download.req(AREP, Area Code, Add Offset, Data)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Download.req(AREP, Area Code, Add Offset, Data) | RUN |
| 63 | RUN | MMAC2_Download.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Download.cnf(+)(AREP) | RUN |
| 64 | RUN | MMAC2_Download.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Download.cnf(-)(AREP, Status) | RUN |
| 65 | RUN | FSPMM2_Upload.req(AREP, Area Code, Add Offset, Data Len)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Upload.req(AREP, Area Code, Add Offset, Data Len) | RUN |
| 66 | RUN | MMAC2_Upload.cnf(+)(Rem_Add, Data)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Upload.cnf(+)(AREP, Data) | RUN |
| 67 | RUN | MMAC2_Upload.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Upload.cnf(-)(AREP, Status) | RUN |
| 68 | RUN | FSPMM2_Act_Param.req(AREP, Area Code, Activate)<br>/AREP.AR_Type = MM1<br>=><br>MMAC2_Act_Param.req(AREP, Area Code, Activate) | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 69 | RUN | MMAC2_Act_Param.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Act_Param.cnf(+)(AREP) | RUN |
| 70 | RUN | MMAC2_Act_Param.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Act_Param.cnf(-)(AREP, Status) | RUN |
| 71 | RUN | FSPMM2_Act_Param_Brct.req(AREP, Area Code)<br>/AREP.AR_Type = MM2<br>=><br>MMAC2_Act_Param_Brct.req(AREP, Area Code, Activate) | RUN |
| 72 | RUN | MMAC2_Act_Param_Brct.cnf(+)(Rem_Add)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Act_Param_Brct.cnf(+)(AREP) | RUN |
| 73 | RUN | MMAC2_Act_Param_Brct.cnf(-)(Rem_Add, Status)<br>/SetContext(Rem_add, Service) = TRUE<br>=><br>FSPMM2_Act_Param_Brct.cnf(-)(AREP, Status) | RUN |
| 74 | tout état | DMPMM2_SYCL_Time_Event.cnf (Send Delay Time, Status)<br>=><br>FSPMM2_SYCL_Time_Event.cnf (AREP, Send Delay Time, Status) | SAME |
| 75 | tout état | FSPMM2_SYCL_Time_Event.req (AREP)<br>=><br>DMPMM2_SYCL_Time_Event.req | SAME |
| 76 | tout état | FSPMM2_SYCL_Clock_Value.req (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status)<br>=><br>DMPMM2_SYCL_Clock_Value.req (Clock Value Time Event, Clock Value previous TE, Clock Value Status) | SAME |
| 77 | tout état | DMPMM2_SYCL_Clock_Value.cnf (Status)<br>=><br>FSPMM2_SYCL_Clock_Value.cnf (AREP, Status) | SAME |
| 78 | tout état | DMPMM2_SYCL_Clock_Value.ind (Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg)<br>=><br>FSPMM2_SYCL_Clock_Value.ind (AREP, Clock Value Time Event, Clock Value previous TE, Clock Value Status, Receive Delay Time, Src add Clk msg) | SAME |
| 79 | tout état | FSPMM2_Reset DP master Cl2 .req<br>=><br>StoreNumberIssuedResets()<br>FSPMM2_UserReset = TRUE<br>DMPMM2_Reset.req<br>MMAC2_Reset.req<br>MSAC2M_Reset.req(First_MS2_AREP ..Last_MS2_AREP) | WAIT-RESET |
| 80 | tout état | DMPMM2_Fault.ind<br>=><br>StoreNumberIssuedResets()<br>FSPMM2_UserReset = FALSE<br>DMPMM2_Reset.req<br>MMAC2_Reset.req<br>MSAC2M_Reset.req(First_MS2_AREP ..Last_MS2_AREP)<br>FSPMM2 DP master Cl2 Fault.ind | WAIT-RESET |
| 81 | tout état | MMAC2_Fault.ind<br>=><br>StoreNumberIssuedResets()<br>FSPMM2_UserReset = FALSE<br>DMPMM2_Reset.req<br>MMAC2_Reset.req<br>MSAC2M_Reset.req(First_MS2_AREP ..Last_MS2_AREP)<br>FSPMM2 DP master Cl2 Fault.ind | WAIT-RESET |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 82 | WAIT-RESET | DMPMM2_Reset.cnf<br>=><br>no action | WAIT-RESET |
| 83 | WAIT-RESET | MMAC2_Reset.cnf<br>=><br>no action | WAIT-RESET |
| 84 | WAIT-RESET | MSAC2M_Reset.cnf (CREP)<br>/IsLastWaitingReset()=FALSE<br>=><br>no action | WAIT-RESET |
| 85 | WAIT-RESET | MSAC2M_Reset.cnf (CREP)<br>/IsLastWaitingReset()=TRUE<br>AND FSPMM2_UserReset = TRUE<br>=><br>FSPMM2_Reset DP master CI2 .cnf | PON |
| 86 | WAIT-RESET | MSAC2M_Reset.cnf (CREP)<br>/IsLastWaitingReset()=TRUE<br>AND FSPMM2_UserReset = FALSE<br>=><br>no action | PON |
| 87 | RUN | FSPMM2_Initiate_Load.req ( AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, User Specific )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Initiate_Load<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=10+sizeof(User Specific)<br>Data.w.IL.Extended_Function_Num:=Initiate_Load<br>Data.w.IL.LR_Index:=LR Index<br>Data.w.IL.Load_Type:=Load Type<br>Data.w.IL.Load_Image_Size:=Load Image Size<br>Data.w.IL.User_Specific:=User Specific<br>Data.w.IL.Intersegment_Request_Timeout:=Intersegment Request Timeout<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 88 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Initiate_Load<br>=><br>Slot_Number:=Current Slot Number[AREP]<br>Index:=255<br>Length:=10<br>MSAC2M_Read.req(CREP, Slot Number, Index, Length)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 89 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Initiate_Load<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Actual LR Size:=Data.r.IL.Actual_LR_Size<br>Max Response Delay:=Data.r.IL.Max_Response_Delay<br>Max Segment Length:=Data.r.IL.Max_Segment_Length<br>User Specific:=Data.r.IL.User_Specific<br>FSPMM2_Initiate_Load.cnf(+) ( AREP, Actual LR Size, Max Response Delay, Max Segment Length, User Specific )<br>LR_State[AREP]:=W-LR-REQ | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 90 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Initiate_Load<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Initiate_Load.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 91 | RUN | FSPMM2_Pull_Segment.req ( AREP, Slot Number, LR Index, Segment Length )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Pull<br>Current LR Index[AREP]:=LR Index<br>Index:=255<br>Length:=Segment Length<br>MSAC2M_Read.req(CREP, Slot Number, Index, Length)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 92 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Pull<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Segment Number:=Data.PULL.Sequence Number<br>More Follows:=Data.PULL.Options.More Follows<br>Data:=Data.PULL.Region Data<br>Segment Length:=Length-6<br>FSPMM2_Pull_Segment.cnf(+) ( AREP, Segment Length, Segment Number, More Follows, Data )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 93 | RUN | MSAC2M_Read.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(Service, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Pull<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Pull_Segment.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 94 | RUN | FSPMM2_Push_Segment.req ( AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Push<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=6+Segment Length<br>Data.PUSH.Extended_Function_Num:=Push<br>Data.PUSH.LR_Index:=LR Index<br>Data.PUSH.Segment_Number:=Segment Number<br>Data.PUSH.Option.More_Follows:=More Follows<br>Data.PUSH.Data:=Data<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 95 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Push<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Segment Number:=Data.PUSH.Sequence Number<br>More Follows:=Data.PUSH.Options.More Follows<br>Data:=Data.PUSH.Region Data<br>Segment Length:=Length-6<br>FSPMM2_Push_Segment.cnf(+) ( AREP)<br>LR_State[AREP]:=W-LR-REQ | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 96 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Push<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Push_Segment.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 97 | RUN | FSPMM2_Terminate_Load.req ( AREP, Slot Number, LR Index )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Terminate_Load<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=6<br>Data.TL.Extended_Function_Num:=Terminate_Load<br>Data.TL.LR_Index:=LR Index<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 98 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Terminate_Load<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Terminate_Load.cnf(+) ( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 99 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Terminate_Load<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Terminate_Load.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 100 | RUN | FSPMM2_Call.req ( AREP, Slot Number, Entity Number, FI Index, Execution Argument)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Call<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.w.CALL.Extended_Function_Num:=Call<br>Data.w.CALL. Entity_Number:= Entity_Number<br>Data.w.CALL.FI_Index:=FI Index<br>Data.w.CALL.Execution_Argument:=Execution Argument<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 101 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Call<br>=><br>Slot_Number:=Current Slot Number[AREP]<br>Index:=255<br>Length:=AREP.Max_PDU_Length<br>MSAC2M_Read.req(CREP, Slot Number, Index, Length)<br>LR_State[AREP]:=W-LR-CNF | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 102 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Call<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Result Argument:=Data.r.CALL.Result_Argument<br>FSPMM2_Call.cnf(+) ( AREP, Result Argument )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 103 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Call<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Call.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 104 | RUN | FSPMM2_Start.req ( AREP, Slot Number, FI Index, Execution Argument)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Start<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.FIS.Extended_Function_Num:=Start<br>Data.FIS.FI_Index:=FI Index<br>Data.FIS.Execution_Argument:=Execution Argument<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 105 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Start<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Start.cnf(+)( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 106 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Start<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Start.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 107 | RUN | FSPMM2_Stop.req ( AREP, Slot Number, FI Index)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Stop<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.FIS.Extended_Function_Num:=Stop<br>Data.FIS.FI_Index:=FI Index<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 108 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Stop<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Stop.cnf(+) ( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 109 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Stop<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Stop.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 110 | RUN | FSPMM2_Resume.req ( AREP, Slot Number, FI Index, Execution Argument)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Resume<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4+sizeof(Execution Argument)<br>Data.FIS.Extended_Function_Num:=Resume<br>Data.FIS.FI_Index:=FI Index<br>Data.FIS.Execution_Argument:=Execution Argument<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 111 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Resume<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Resume.cnf(+) ( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 112 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Resume<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Resume.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 113 | RUN | FSPMM2_Reset.req ( AREP, Slot Number, FI Index)<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Reset<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.FIS.Extended_Function_Num:=Reset<br>Data.FIS.FI_Index:=FI Index<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 114 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Reset<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FSPMM2_Reset.cnf(+) ( AREP )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 115 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Reset<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Reset.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 116 | RUN | FSPMM2_Get_FI_State.req ( AREP, Slot Number, FI Index )<br>/LR_State[AREP]==W-LR-REQ<br>&AREP.AR_Type=MS2<br>=><br>Current Extended Function Num[AREP]:=Get_State<br>Current Slot Number[AREP]:=Slot Number<br>Index:=255<br>Length:=4<br>Data.w.STATE.Extended_Function_Num:=Get_State<br>Data.w.STATE.FI_Index:=FI Index<br>MSAC2M_Write.req(CREP, Slot Number, Index, Length, Data)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 117 | RUN | MSAC2M_Write.cnf(+)(CREP, Length)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Get_State<br>=><br>Slot_Number:=Current Slot Number[AREP]<br>Index:=255<br>Length:=5<br>MSAC2M_Read.req(CREP, Slot Number, Index, Length)<br>LR_State[AREP]:=W-LR-CNF | RUN |
| 118 | RUN | MSAC2M_Read.cnf(+)(CREP, Length, Data)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Get_State<br>=><br>Current Extended Function Num[AREP]=No_Service<br>FI State:=Data.r.STATE.State<br>FSPMM2_Get_FI_State.cnf (+) ( AREP, FI State )<br>LR_State[AREP]:=W-LR-REQ | RUN |
| 119 | RUN | MSAC2M_Write.cnf(-)(CREP, Error Decode, Error Code 1, Error Code 2)<br>/LR_State[AREP]==W-LR-CNF<br>&SetContext(CREP, Service)=TRUE &&<br>Current Extended Function Num[AREP]=Get_State<br>=><br>Current Extended Function Num[AREP]=No_Service<br>Error Code:=Error_Code_1<br>FSPMM2_Get_State.cnf(-) ( AREP, Error Code )<br>LR_State[AREP]:=W-LR-REQ | RUN |

### 8.3.4 Fonctions

Le Tableau 44 contient les fonctions utilisées par le FSPMM2, leurs arguments et leurs descriptions.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 44 – Fonctions utilisées par le FSPMM2**

| Nom de fonction | Description |
|---|---|
| SetContext(Rem_Add/CREP, Service) | Cette fonction vérifie s'il existe une entrée saisie pour les entrées Rem_Add /CREP et Service dans l'ARL et la CRL correspondante du maître DP (Classe 2).<br>A) S'il existe une entrée,<br>elle retourne TRUE et règle contexte local selon le CREP et l'AREP courants.<br>B) Autrement, elle retourne FALSE. |
| IsLastWaitingReset() | Cette fonction retourne TRUE si toutes les confirmations de service Reset sont reçues; autrement, elle retourne FALSE. |
| StoreNumberIssuedResets() | Cette fonction stocke dans une variable locale le nombre de demandes de service Reset émises. |
| IsLastWaitingMInit() | Cette fonction retourne TRUE si toutes les confirmations de service MInit sont reçues; autrement, elle retourne FALSE. |
| StoreNumberIssuedMInit() | Cette fonction stocke dans une variable locale le nombre de demandes de service MInit émises. |

# 9 Diagramme d'état de protocol de mapping d'application (ARPM)

## 9.1 MSCY1S

### 9.1.1 Définitions des primitives

#### 9.1.1.1 Primitives échangées entre MSCY1S et FSPMS

Le Tableau 45 montre les primitives de service, y compris leurs paramètres associés, émises par le FSPMS et reçues par le MSCY1S.

**Tableau 45 – Primitives émises par le FSPMS vers le MSCY1S**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| SInit MS0.req | FSPMS | (aucun) | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3. |
| Reset.req | FSPMS | (aucun) | |
| Abort.req | FSPMS | (aucun) | |
| Application Ready.req | FSPMS | (aucun) | |
| Check User Prm Result.req | FSPMS | Prm_OK | |
| Check Ext User Prm Result.req | FSPMS | Ext_Prm_OK | |
| Check Cfg Result.req | FSPMS | Cfg_OK,<br>Input Data Len,<br>Output Data Len | |
| Set Cfg.req | FSPMS | Cfg Data | |
| Set Slave Diag.req | FSPMS | Ext Diag Flag,<br>Ext Diag Overflow,<br>Ext Diag Data | |
| Set Input.req | FSPMS | Input Data | |
| Get Output.req | FSPMS | (aucun) | |

Le Tableau 46 montre les primitives de service, y compris leurs paramètres associés, émises par le MSCY1S et reçues par le FSPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 46 – Primitives émises par le MSCY1S vers le FSPMS**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| SInit MS0.cnf | MSCY1S | (aucun) | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3. |
| Reset.cnf | MSCY1S | (aucun) | |
| Check User Prm Result.cnf(+) | MSCY1S | (aucun) | |
| Check User Prm Result.cnf(-) | MSCY1S | Status | |
| Check Ext User Prm Result.cnf(+) | MSCY1S | (aucun) | |
| Check Ext User Prm Result.cnf(-) | MSCY1S | Status | |
| Check Cfg Result.cnf(+) | MSCY1S | (aucun) | |
| Check Cfg Result.cnf(-) | MSCY1S | Status | |
| Set Cfg.cnf(+) | MSCY1S | (aucun) | |
| Set Cfg.cnf(-) | MSCY1S | (aucun) | |
| Set Slave Diag.cnf(+) | MSCY1S | (aucun) | |
| Set Slave Diag.cnf(-) | MSCY1S | (aucun) | |
| Set Input.cnf(+) | MSCY1S | (aucun) | |
| Set Input.cnf(-) | MSCY1S | (aucun) | |
| Get Output.cnf(+) | MSCY1S | Output Data,<br>Clear Flag,<br>New Flag | |
| Get Output.cnf(-) | MSCY1S | (aucun) | |
| Start.ind | MSCY1S | Actual Enabled Alarms,<br>Alarm Sequence,<br>Alarm Limit | |
| Stop.ind | MSCY1S | (aucun) | |
| Fault.ind | MSCY1S | (aucun) | |
| Set Slave Add.ind | MSCY1S | New Slave Add,<br>Ident Number,<br>No Add Chg,<br>Rem Slave Data | |
| Check User Prm.ind | MSCY1S | User Prm Data (données Prm utilisateur) | |
| Check Ext User Prm.ind | MSCY1S | Ext User Prm Data | |
| Check Cfg.ind | MSCY1S | Check Cfg Mode,<br>Cfg Data | |
| New Output.ind | MSCY1S | Clear Flag | |
| SYNCH Event.ind | MSCY1S | Status | |
| Global Control.ind | MSCY1S | Clear Command,<br>Sync Command,<br>Freeze Command,<br>Group Select | |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

#### 9.1.1.2 Paramètres des primitives de MSCY1S

Les paramètres utilisés avec les primitives échangées entre le MSCY1S et le FSPMS sont décrits dans "Définition des services de la FAL" (voir la CEI 61158-5-3).

### 9.1.2 Description de diagramme d'états

Le diagramme d'états MSCY1S gère les services suivants émis par le maître DP (Classe 1):

diagnostic,

paramétrisation,

configuration,

échange de données (cyclique).

Le diagramme d'états MSCY1S démarre (Start.req/cnf) et arrête (Stop.req/cnf) le diagramme d'états MSAC1S qui gère des services cycliques. Le diagramme d'états MSAC1S envoie une Abort.ind vers le diagramme d'états MSCY1S qui ferme aussi la connexion cyclique. Le diagramme d'états MSCY1S informe le FSPMS que le MSAC1S et le MSCY1S ont été démarrés (Start.ind) ou arrêtés (Stop.ind).

Si un Esclave ne met pas en œuvre les structures de paramétrisation DPV1_Status 1 à 3, les diagrammes d'états MSAL1S et MSAC1S, il convient de laisser vides les macros suivantes. Dans ce cas, la variable Operation_Mode reste à l'état V0:

PV0V0 doit retourner TRUE

OPERATION_MODE_OK doit retourner TRUE

NOPRMCMD doit retourner TRUE

ISO_MODE_OK doit retourner TRUE

STOP_ISOM

STOP_ASM

CHECK_ALARM_START

SET_OPERATION_MODE

SET_ALARM_CHKCFGM

EXE_PRM_CMD

START_C1

STOP_C1

START_C1_CON

STOP_C1_CON

**Règles pour vérifier les données de paramétrisation**

Les règles selon le Tableau 47 doivent être utilisées pour vérifier les octets DPV1_Status_1 à DPV1_Status_3 issus de la Set_Prm-REQ-PDU avec les capacités locales de l'esclave DP.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 47 – Règles pour la vérification de DPV1_Status_1, DPV1_Status_2 et DPV1_Status_3**

| Bit(s) dans Set_Prm-REQ-PDU | Règles pour la vérification | |
|---|---|---|
| Bits réservés dans le champ DPV1_Status_1 ( Bit 0, Bit 1, Bit 3, Bit 4) | Ces bits ne doivent pas être vérifiés (sans influence). => okay (c'est-à-dire: correct) pour toute combinaison | |
| Bit WD_Base_1ms dans le champ DPV1_Status_1 | = FALSE(0):<br>=> okay | = TRUE (1) & WD_Base_1ms_supp = 1:<br>=> okay<br>= TRUE (1) & WD_Base_1ms_supp = 0:<br>=> okay, mais l'esclave DP doit charger son WD-Timer local avec une valeur corrigée (basée sur la prochaine valeur possible avec la base de 10 ms) |
| Bit Publisher_Enable dans le champ DPV1_Status_1 | = FALSE(0):<br>=> okay | = TRUE(1) & Publisher_supp = 1:<br>=> okay<br>= TRUE(1) & Publisher_supp = 0:<br>=> Prm_Fault |
| Bit Fail_Safe dans le champ DPV1_Status_1 | = FALSE(0) & Fail_Safe_required = 1:<br>=> Prm_Fault<br>= FALSE(0) & Fail_Safe_required = 0:<br>=> okay | = TRUE(1) & Fail_Safe_supp = 1:<br>=> okay<br>= TRUE(1) & Fail_Safe_supp = 0:<br>=> Prm_Fault |
| Bit DPV1_Enable dans le champ DPV1_Status_1 | = FALSE(0) & C1_Read_Write_required=0:<br>=> okay<br>= FALSE(0) & C1_Read_Write_required=1:<br>=> Prm_Fault | = TRUE(1) & C1_Read_Write_supp = 0:<br>=> Prm_Fault<br>= TRUE(1) & C1_Read_Write_supp = 1:<br>=> okay, ouvrir la relation d'applications MS1 |
| Bit Check_Cfg_Mode dans le champ DPV1_Status_2 | = 0 (vérification stricte):<br>=> okay<br>Le processus d'application doit vérifier les données de la Chk_Cfg-REQ-PDU suivante de façon stricte (égalité). | = 1 (vérification plus souple):<br>=> okay;<br>Le processus d'application doit vérifier les données de la Chk_Cfg-REQ-PDU suivante de façon plus souple (compatibilité). |
| Bit 1 réservé dans le champ DPV1_Status_2 | = 0:<br>=> okay | = 1:<br>=> Prm_Fault |
| Bits Enable_xx_Alarm dans le champ DPV1_Status_2 | = FALSE(0) & xx_Alarm_required = 1 & DPV1_Status_1.DPV1_Enable = FALSE (0):<br>=> Prm_Fault<br>= FALSE(0) & xx_Alarm_required = 0 & DPV1_Status_1.DPV1_Enable = FALSE (0):<br>=> okay<br>= FALSE(0) & xx_Alarm_required = 1 & DPV1_Status_1.DPV1_Enable = TRUE (1):<br>=> Prm_Fault<br>= FALSE(0) & xx_Alarm_required = 0 & DPV1_Status_1.DPV1_Enable = TRUE (1):<br>=> okay | =TRUE(1) & DPV1_Status_1.DPV1_Enable = FALSE(0):<br>=> Prm_Fault<br>=TRUE(1) & DPV1_Status_1.DPV1_Enable = TRUE (1):<br>=> okay |
| Bits Alarm_Mode dans le champ DPV1_Status_3 | 0<br><br>=> okay | <>0 & Alarm_Sequence_Mode_count<>0:<br>=> okay<br><>0 & Alarm_Sequence_Mode_count=0:<br>=> Prm_Fault |
| Bit Prm_Structure dans le champ DPV1_Status_3 | = FALSE(0) & Prm_Structure_required=FALSE:<br>=>okay<br>= FALSE(0) & Prm_Structure_required=TRUE:<br>=> Prm_Fault | = TRUE(1) & Prm_Structure_supp=TRUE:<br>=>okay<br><br>= TRUE(1) & Prm_Structure_supp=FALSE:<br>=> Prm_Fault |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| Bit(s) dans Set_Prm-REQ-PDU | Règles pour la vérification | |
|---|---|---|
| Bit IsoM_Req dans le champ DPV1_Status_3 | = FALSE(0) & Isochronous_Mode_required=FALSE: =>okay<br>= FALSE(0) & Isochronous_Mode_required=TRUE: => Prm_Fault | = TRUE(1) & (Isochronous_Mode_supp = TRUE & DPV1_Status_1.Fail_Safe = TRUE & Group_Ident< 0x80 & Freeze_Req=0 & Sync_Req=0): => okay<br>TRUE(1) & !(Isochronous_Mode_supp = TRUE & DPV1_Status_1.Fail_Safe = TRUE & Group_Ident< 0x80 & Freeze_Req=0 & Sync_Req=0): => Prm-Fault |
| Bit PrmCmd dans le champ DPV1_Status_3 | = FALSE(0) & PrmCmd_required=FALSE: =>okay<br>= FALSE(0) & PrmCmd_required=TRUE:<br>=> Prm_Fault | = TRUE(1) & (PrmCmd_supp = TRUE): => okay<br>TRUE(1) & (PrmCmd_supp = FALSE): => Prm-Fault |
| Bits réservés (Bit 5 et Bit 6) dans le champ DPV1_Status_3 | = 0: => okay | 1: => Prm_Fault |
| NOTE Le tableau ci-dessus donne une vue d'ensemble complète de toutes les vérifications qui appartiennent au paramètre. Cependant; certaines d'entre elles peuvent être vérifiées en d'autres endroits du diagramme d'états. | | |

### Comportement isochrone

Les dispositifs qui prennent en charge le fonctionnement isochrone doivent positionner les attributs AR concernés selon la configuration. Le bloc pour les paramètres isochrones appartient à l'application et fait partie des données utilisateur dans la Set_Prm-REQ-PDU ou dans la Set_Ext_Prm-REQ-PDU. Le maître DP met l'esclave DP dans le fonctionnement isochrone en mettant le bit IsoM_Req à TRUE. Lorsque le MSCY1S reçoit la Set_Prm-REQ-PDU, il doit vérifier la condition suivante:

Si (Isochronous Mode = TRUE), alors Failsafe doit être mis et aucun groupe ne doit être mis concernant la Set_Prm-REQ-PDU; autrement, les esclaves DP doivent générer un Prm_Fault et doivent quitter l'état d'échange de données "Data Exchange". Les esclaves DP qui ne prennent pas en charge le fonctionnement isochrone peuvent être intégrés dans un système de fonctionnement isochrone en sélectionnant “Group_8” pour ces esclaves DP seulement et en utilisant les commandes Sync et Freeze.

Si la condition ci-dessus est remplie, l'esclave DP fonctionne de façon isochrone. En plus de l'échange cyclique normal de données, le MSCY1S génère une SYNCH Event.ind avec le Status “IsoM Start” avec la réception de la première Global_Control-REQ-PDU, “IsoM SYNCH” avec les Global_Control-REQ-PDU suivantes et “IsoM Stop” lors de la réception de la Global_Control-REQ-PDU avec la commande de commandes “Clear” ou en quittant le cycle d'échange de données. Ce service déclenche la PLL (Phase-Lock Loop, boucle à verrouillage de phase) du processus d'application. L'accès correct avec Get Input et Set Output du point de vue de l'application est surveillé avec l'aide de la PLL.

### Comportement de redondance

Les dispositifs qui prennent en charge la redondance d'esclave doivent prendre en charge la PrmCmd. Cela permet à un maître DP (Classe 1) d'utiliser un esclave DP comme Primary ou Backup. En outre, le maître DP (Classe 1) peut arrêter et démarrer MS1 après une commutation du maître DP. Le MSCY1S fournit le mécanisme de base pour prendre en charge une structure de redondance dans l'Application.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Variables locales**

**Act_Ref**
Compteur pour numéros de référence pour les mises à jour de diagnostics.

**Act_Cnt**
Stockage pour numéro de référence des utilisateurs

**Ref_Cnt**
Stockage d'Act_Ref au diagnostic d'utilisateurs

**B Sync**
(Octet-String)

Stockage intermédiaire pour les sorties si l'esclave DP est exploité en mode Sync.

**B Output**
(Octet-String)

Données de sortie de l'esclave DP qui peuvent être récupérées par le service Get Output. Les exigences pour la cohérence telle que décrite dans les Cfg_Data doivent être prises en considération.

**B Freeze**
(Octet-String)

Stockage intermédiaire pour données d'entrée

**B Input**
(Octet-String)

Selon le type (voir la description de Cfg_Data), les données doivent être chargées de façon cohérente par le service Set Input ou peuvent être saisies librement.

**B Real_Cfg**
(Octet-String)

La configuration qui a été précédemment définie pour le dispositif ou la configuration qui a été déterminée dans le démarrage par le dispositif. La structure des octets individuels correspond à la structure de Cfg_Data (->Chk_Cfg).

**B User Prm**
(Octet-String)

Stockage intermédiaire pour données User Prm.

**B Ext User Prm**
(Octet-String)

Stockage intermédiaire pour données Ext User Prm.

**B Cfg**
(Octet-String)

Stockage intermédiaire pour données Cfg

**Diag**
(Octet-String)

Stockage intermédiaire pour données Diag.

**Ext Diag Data**
(Octet-String)

Withdrawn
IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

Stockage intermédiaire pour les Ext Diag Data, l'établissement de diagnostic par l'utilisateur.

**Ext Diag Flag**
(Boolean)

Stockage intermédiaire pour Ext Diag Flag, fanion qui indique un diagnostic utilisateur important .

**Clear Command**
**Sync Command**
**Freeze Command**
Stockage intermédiaire pour commandes de commandes globales.

**Station_Address**
(Unsigned8)

Propre adresse de poste qui peut être établie par le service "Set_Slave_Add" ou qui peut être stockée localement.

**Check_Prm_Add**
(Unsigned8)

Adresse de poste de maîtres qui a invoquée le dernier servie "Set_Prm ".

**Check_Ext_Prm_Add**
(Unsigned8)

Adresse de poste de maîtres qui a invoquée le dernier servie "Set_Ext_Prm".

**Check_Cfg_Add**
(Unsigned8)

Adresse de poste de maîtres qui a invoquée le dernier servie "Chk_Cfg".

**Output Data Len**
(Unsigned8)

Indique le nombre d'octets de sortie qui sont réellement utilisés.

**Input Data Len**
(Unsigned8)

Indique le nombre d'octets d'entrée qui sont réellement utilisés.

**Real_No_Add_Chg**
(Boolean)

Indique si un changement d'adresse est possible (FALSE) ou non. Peut être définie de façon statique ou stockée.

**Active_Groups**
(Unsigned8)

Contient le Group_Ident tel qu'établi par le biais de Set_Prm. Elle est utilisée à la Global_Control pour la décision si le poste est adressé.

**Diag_Flag**
(Boolean)

Ce fanion indique que le Maître doit être informé par une Reply-Update à haute priorité qu'un changement s'est produit dans les informations de diagnostic.

**Sync_Supp**
(Boolean)

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

Indique que l'esclave DP prend effectivement en charge le mode Sync.

**Freeze_Supp**
(Boolean)

Indique que l'esclave DP prend effectivement en charge le mode Freeze.

**Operation_Mode**
(Boolean)

Fanion local qui stocke l'Operation_Mode établi par le service Set_Prm . Operation_Mode=V0 signifie que l'Esclave ne prend pas en charge les services acycliques sur MSAC1 et les alarmes.

**Prm_Pending**
(Unsigned8)

Fanion local qui indique que l'utilisateur traite encore le dernier appel de la fonction "Check UserPrm ".

**Ext_Prm_Pending**
(Unsigned8)

Fanion local qui indique que l'utilisateur traite encore le dernier appel de la fonction "Check Ext_Prm".

**Cfg_Pending**
(Unsigned8)

Fanion local qui indique que l'utilisateur traite encore le dernier appel de la fonction "Check Cfg".

**Input_Pending**
(Unsigned8)

Fanion local qui indique que l'utilisateur n'a pas encore établi les données d'entrée.

**New Output**
(Unsigned8)

Fanion local qui indique qu'il y a des données de sortie (Output data) disponibles qui n'ont pas encore été récupérées avec "Get Output".

**DX_Entered**
(Boolean)

Fanion local qui est mis après que le diagramme d'états MSCY1S s'est mis dans le mode DATA-EXCH et au moins un service Data_Exchange a été parachevé.

**Start_State**
(Unsigned8)

Variable locale qui est utilisée pour stocker l'état du diagramme d'états MSAC1S.

Valeurs possibles pour Start_State:

Idle MSAC1S est fermé.

Run MSAC1S est ouvert.

B Start_Operation est lancé.

E Stop_Operation est lancé.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

BE Start_Operation est lancé, Stop_Operation est en file d'attente.

EB Stop_Operation est lancé, Start_Operation est en file d'attente.

BEB Start_Operation est lancé, Stop_Operation (premier) et Start_Operation (second) sont en file d'attente.

**Safe_State**
(Boolean)

Indique si les sorties doivent être commutées vers l'état de sécurité.

**Chk Cfg Mode**
(Boolean)

Indique comment vérifier les données Cfg Data.

**First Synch**
(Boolean)

Cette variable locale est FALSE si le mode Isochronous a été activé et au moins une Global_Control-REQ-PDU avec Group Select (Group_8) a été reçue.

### 9.1.3 Table d'états de MSCY1S

Le Tableau 48 contient la description complète du diagramme d'états MSCY1S.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 48 – Table d'états de MSCY1S**

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 1 | POWER-ON | MSCY1S SInit MS0.req ()<br>=><br>Diag.Ext_Diag_Data:= NIL<br>Diag.Ext Diag Flag:= FALSE<br>B Real Cfg:= Real Cfg Data<br>Chk Cfg Mode:= FALSE<br>Sync Supp:= Sync Supported<br>Freeze Supp:= Freeze Supported<br>Diag.Ident Number:= Ident Number<br>Real No Add Chg:= No Add Chg<br>Act_Ref:= 0<br>Act_Cnt:= 0<br>Ref_Cnt:= 0<br>DMPMS Slave Init.req () | WAIT-INI-CON |
| 2 | POWER-ON | MSCY1S Abort.req ()<br>=><br>ignore | POWER-ON |
| 3 | POWER-ON | MSCY1S Set Slave Diag.req ( Ext Diag Flag, Ext Diag Overflow, Ext Diag Data, Reference )<br>=><br>Act_Cnt:= Reference<br>MSCY1S Set Slave Diag.cnf(-) (Reference:= Act_Cnt) | POWER-ON |
| 4 | POWER-ON | MSCY1S Set Cfg.req ( Cfg Data )<br>=><br>MSCY1S Set Cfg.cnf(-) () | POWER-ON |
| 5 | POWER-ON | MSCY1S Get Output.req ()<br>=><br>MSCY1S Get Output.cnf(-) () | POWER-ON |
| 6 | POWER-ON | MSCY1S Set Input.req ( Input Data )<br>=><br>MSCY1S Set Input.cnf(-) () | POWER-ON |
| 7 | POWER-ON | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>=><br>Status:= Reset<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | POWER-ON |
| 8 | POWER-ON | MSCY1S Application Ready.req()<br>=><br>ignore | POWER-ON |
| 9 | POWER-ON | MSCY1S Check User Prm Result.req ( Prm_OK )<br>=><br>Status:= Reset<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 10 | POWER-ON | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>=><br>Status:= Reset<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | POWER-ON |
| 11 | WAIT-INI-CON | DMPMS Slave Init.cnf ()<br>=><br>Operation Mode:= V0<br>Start_State:= Idle<br>Safe_State:=TRUE<br>Diag.Prm_Req:= TRUE<br>Diag.Cfg_Fault:= FALSE<br>Diag.Prm_Fault:= FALSE<br>Diag.Not_Supported:= FALSE<br>Diag.Master_Add:= Invalid<br>Diag.WD_On:= FALSE<br>Diag.Station_Not_Ready:= TRUE<br>Diag.Sync_Mode:= FALSE<br>Diag.Freeze_Mode:= FALSE<br>Diag.Stat_Diag:= FALSE<br>Diag.Ext_Diag_Overflow:= FALSE<br>B User Prm:= NIL<br>B Cfg:= NIL<br>Diag_Flag:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>Prm Structure = FALSE<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Get Cfg Upd.req ( Cfg Data )<br>MSCY1S SInit MS0.cnf () | WAIT-PRM |
| 12 | WAIT-PRM | MSCY1S Abort.req ()<br>=><br>ignore | WAIT-PRM |
| 13 | WAIT-PRM | MSCY1S Set Slave Diag.req ( Ext Diag Flag, Ext Diag Overflow, Ext Diag Data, Reference )<br>=><br>Diag.Ext Diag Data:= Ext Diag Data<br>Diag.Ext Diag Flag:= Ext Diag Flag<br>Diag.Ext Diag Overflow:= Ext Diag Overflow<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>Act_Cnt:= Reference<br>Ref_Cnt:= Act_Ref<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>MSCY1S Set Slave Diag.cnf(+) (Reference:= Act_Cnt ) | WAIT-PRM |
| 14 | WAIT-PRM | MSCY1S Set Cfg.req ( Cfg Data )<br>=><br>DMPMS Get Cfg Upd.req ( Cfg Data )<br>MSCY1S Set Cfg.cnf(+) () | WAIT-PRM |
| 15 | WAIT-PRM | MSCY1S Get Output.req ()<br>=><br>MSCY1S Get Output.cnf(-) () | WAIT-PRM |
| 16 | WAIT-PRM | MSCY1S Set Input.req ( Input Data )<br>=><br>MSCY1S Set Input.cnf(-) () | WAIT-PRM |
| 17 | WAIT-PRM | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | WAIT-PRM |
| 18 | WAIT-PRM | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 2<br>=><br>Status:= New Prm<br>Prm_Pending:= 1<br>User Parameter Data:= B User Prm<br>MSCY1S Check User Prm Result.cnf(-) ( Status )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 19 | WAIT-PRM | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1<br>=><br>Prm_Pending:= 0<br>MSCY1S Check User Prm Result.cnf(+)() | WAIT-PRM |
| 20 | WAIT-PRM | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | WAIT-PRM |
| 21 | WAIT-PRM | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 2<br>=><br>Status:= New Cfg<br>Cfg_Pending:= 1<br>Cfg Data:= B Cfg<br>MSCY1S Check Cfg Result.cnf(-) ( Status )<br>MSCY1S Check Cfg.ind ( Cfg Data ) | WAIT-PRM |
| 22 | WAIT-PRM | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1<br>=><br>Cfg_Pending:= 0<br>MSCY1S Check Cfg Result.cnf(+) () | WAIT-PRM |
| 23 | WAIT-PRM | MSCY1S Application Ready.req()<br>=><br>ignore | WAIT-PRM |
| 24 | WAIT-PRM | MSAC1S Abort.ind()<br>=><br>ignore | WAIT-PRM |
| 25 | WAIT-PRM | MSAC1S Start.cnf()<br>=><br>START_C1_CON | WAIT-PRM |
| 26 | WAIT-PRM | MSAC1S Stop.cnf()<br>=><br>Start_State:= Idle | WAIT-PRM |
| 27 | WAIT-PRM | DMPMS Set Slave Add.ind(New Slave Add, Ident Number, No Add Chg, Rem Slave Data)<br>/Real_No_Add_Chg = FALSE && Diag.Ident_Number = Ident_Number<br>&& New_Slave_Address <= 125<br>=><br>MSCY1S Set Slave Add.ind ( New Slave Add, Ident Number, No Add Chg, Rem Slave Data ) | POWER-ON |
| 28 | WAIT-PRM | DMPMS Set Slave Add.ind(New Slave Add, Ident Number, No Add Chg, Rem Slave Data)<br>/Real_No_Add_Chg = TRUE \|\| Diag.Ident_Number <> Ident_Number<br>\|\| New_Slave_Add > 125<br>=><br>ignore | WAIT-PRM |
| 29 | WAIT-PRM | DMPMS Slave Diag.ind(Req Add, Reference)<br>=><br>ignore | WAIT-PRM |
| 30 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = FALSE && Unlock_Req = FALSE<br>=><br>DMPMS Set minTsdr.req ( MinTsdr ) | WAIT-PRM |
| 31 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE<br>=><br>ignore | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 32 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE &&<br>(Ident_Number <> Diag.Ident_Number \|\| !OPERATION_MODE_OK \|\| WD_On = TRUE && (WD_Fact_1=0 \|\| WD_Fact_2=0) )<br>=><br>Diag.Prm_Fault:= TRUE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 33 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE<br>&& Ident_Number = Diag.Ident_Number && OPERATION_MODE_OK &&<br>(WD_On = FALSE \|\| (WD_Fact_1 > 0 && WD_Fact_2 > 0))<br>&& (Freeze_Req = TRUE && Freeze Supp = FALSE \|\| Sync_Req = TRUE<br>&& Sync Supp = FALSE \|\| Prm_Data[1].0, .1, .2 = TRUE)<br>=><br>Diag.Not_Supported:= TRUE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 34 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len < 7<br>=><br>ignore | WAIT-PRM |
| 35 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/PRM_OK && Prm_Pending = 0<br>=><br>Diag.Master_Add:= Req Add<br>Check_Prm_Add:= Req Add<br>FirstSynch:= TRUE<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups:= Group_Ident<br>Diag.Prm_Fault:= FALSE<br>Diag.Prm_Req:= FALSE<br>Diag.Not_Supported:= FALSE<br>Prm_Pending:= 1<br>User Parameter Data:= Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Input_Pending:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-CFG |
| 36 | WAIT-PRM | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/PRM_OK && Prm_Pending > 0<br>=><br>Diag.Master_Add:= Req Add<br>Check_Prm_Add:= Req Add<br>FirstSynch:= TRUE<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups:= Group_Ident<br>Diag.Prm_Fault:= FALSE<br>Diag.Prm_Req:= FALSE<br>Diag.Not_Supported:= FALSE<br>Prm_Pending:= 2<br>B User Prm:= Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Input_Pending:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | WAIT-CFG |
| 37 | WAIT-PRM | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>=><br>ignore | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 38 | WAIT-PRM | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add = Req_Add<br>=><br>ignore | WAIT-PRM |
| 39 | WAIT-PRM | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | WAIT-PRM |
| 40 | WAIT-PRM | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 2<br>=><br>Status:= New Prm<br>Prm_Pending:= 1<br>Ext User Parameter Data:= B Ext User Prm<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status )<br>MSCY1S Check Ext User Prm.ind ( Ext User Parameter Data ) | WAIT-PRM |
| 41 | WAIT-PRM | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1<br>=><br>Ext_Prm_Pending:= 0<br>MSCY1S Check Ext User Prm Result.cnf(+)() | WAIT-PRM |
| 42 | WAIT-CFG | MSCY1S Abort.req ()<br>=><br>Operation_Mode:= V0<br>Diag.Master_Add:= invalid<br>Diag.Prm_Req:= TRUE<br>Diag.Station_Not_Ready:= TRUE<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 43 | WAIT-CFG | MSCY1S Set Slave Diag.req ( Ext Diag Flag, Ext Diag Overflow, Ext Diag Data, Reference )<br>=><br>Diag.Ext Diag Data:= Ext Diag Data<br>Diag.Ext Diag Flag:= Ext Diag Flag<br>Diag.Ext Diag Overflow:= Ext Diag Overflow<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>Act_Cnt:= Reference<br>Ref_Cnt:= Act_Ref<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>MSCY1S Set Slave Diag.cnf(+) (Reference:= Act_Cnt ) | WAIT-CFG |
| 44 | WAIT-CFG | MSCY1S Set Cfg.req ( Cfg Data )<br>=><br>DMPMS Get Cfg Upd.req ( Cfg Data )<br>MSCY1S Set Cfg.cnf(+) () | WAIT-CFG |
| 45 | WAIT-CFG | MSCY1S Get Output.req ()<br>=><br>MSCY1S Get Output.cnf(-) () | WAIT-CFG |
| 46 | WAIT-CFG | MSCY1S Set Input.req ( Input Data )<br>/Input_Pending = FALSE<br>=><br>MSCY1S Set Input.cnf(-) () | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 47 | WAIT-CFG | MSCY1S Set Input.req ( Input Data )<br>/Input_Pending = TRUE && Operation_Mode = V0<br>=><br>Inp Data:= Input Data<br>Diag.Cfg_Fault:= FALSE<br>Diag_Flag:= TRUE<br>Diag.Stat_Diag:= TRUE<br>Diag.Station_Not_Ready:= FALSE<br>DX_Entered:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>MSCY1S Set Input.cnf(+) ()<br>DMPMS Enter.req ( Master Add, Input Data Len, Output Data Len )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | DATA-EXCH |
| 48 | WAIT-CFG | MSCY1S Set Input.req ( Input Data )<br>/Input_Pending = TRUE && Operation_Mode = V1<br>=><br>B Input:= Input Data<br>START_C1<br>MSCY1S Set Input.cnf(+) () | WAIT-CFG |
| 49 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | WAIT-CFG |
| 50 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 2<br>=><br>Status:= New Prm<br>Prm_Pending:= 1<br>User Parameter Data:= B User Prm<br>MSCY1S Check User Prm Result.cnf(-) ( Status )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-CFG |
| 51 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add <> Check_Prm_Add<br>=><br>Status:= Inv Master Add<br>Prm_Pending:= 0<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | WAIT-CFG |
| 52 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Prm_OK = TRUE<br>=><br>Prm_Pending:= 0<br>MSCY1S Check User Prm Result.cnf(+)() | WAIT-CFG |
| 53 | WAIT-CFG | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Prm_OK = FALSE<br>=><br>Prm_Pending:= 0<br>Diag.Prm_Fault:= TRUE<br>Diag.Master_Add:= Invalid<br>Diag.Prm_Req:= TRUE<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>MSCY1S Check User Prm Result.cnf(+)()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 54 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 55 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 2<br>=><br>Status:= New Cfg<br>Cfg_Pending:= 1<br>Cfg Data:= B Cfg<br>MSCY1S Check Cfg Result.cnf(-) ( Status )<br>MSCY1S Check Cfg.ind ( Cfg Data ) | WAIT-CFG |
| 56 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add <> Check_Cfg_Add<br>=><br>Status:= Inv Master Add<br>Cfg_Pending:= 0<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | WAIT-CFG |
| 57 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Input Data Len > 0<br>=><br>Inp Data Len:= Input Data Len<br>Outp Data Len:= Output Data Len<br>Cfg_Pending:= 0<br>Input_Pending:= TRUE<br>MSCY1S Check Cfg Result.cnf(+) () | WAIT-CFG |
| 58 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Operation_Mode = V1 && Input Data Len = 0<br>=><br>Inp Data Len:= 0<br>Outp Data Len:= Output Data Len<br>Inp Data:= NIL<br>Cfg_Pending:= 0<br>START_C1<br>MSCY1S Check Cfg Result.cnf(+) () | WAIT-CFG |
| 59 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Operation_Mode = V0 && Input Data Len = 0<br>=><br>Inp Data Len:= 0<br>Outp Data Len:= Output Data Len<br>Inp Data:= NIL<br>Cfg_Pending:= 0<br>Diag.Cfg_Fault:= FALSE<br>Diag_Flag:= TRUE<br>Diag.Stat_Diag:= TRUE<br>Diag.Station_Not_Ready:= FALSE<br>DX_Entered:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Enter.req<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>MSCY1S Check Cfg Result.cnf(+) () | DATA-EXCH |
| 60 | WAIT-CFG | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = FALSE<br>=><br>Cfg_Pending:= 0<br>Diag.Cfg_Fault:= TRUE<br>Diag.Prm_Req:= TRUE<br>Diag.Master_Add:= Invalid<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>MSCY1S Check Cfg Result.cnf(+) () | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 61 | WAIT-CFG | MSCY1S Application Ready.req()<br>=><br>ignore | WAIT-CFG |
| 62 | WAIT-CFG | MSAC1S Abort.ind()<br>=><br>ignore | WAIT-CFG |
| 63 | WAIT-CFG | MSAC1S Start.cnf()<br>/Start_State <> B<br>=><br>START_C1_CON | WAIT-CFG |
| 64 | WAIT-CFG | MSAC1S Start.cnf()<br>/Start_State = B<br>=><br>Diag.Cfg_Fault:= FALSE<br>Diag_Flag:= TRUE<br>Diag.Stat_Diag:= TRUE<br>Diag.Station_Not_Ready:= FALSE<br>DX_Entered:= FALSE<br>START_C1_CON<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Enter.req<br>DMPMS RD Inp Upd.req ( Inp Data )<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | DATA-EXCH |
| 65 | WAIT-CFG | MSAC1S Stop.cnf()<br>=><br>STOP_C1_CON | WAIT-CFG |
| 66 | WAIT-CFG | MSAC1S Abort.ind()<br>=><br>Operation_Mode:= V0<br>Diag.Master_Add:= invalid<br>Diag.Prm_Req:= TRUE<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 67 | WAIT-CFG | DMPMS Set Slave Add.ind(New Slave Add, Ident Number, No Add Chg, Rem Slave Data)<br>=><br>ignore | WAIT-CFG |
| 68 | WAIT-CFG | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add = Req_Add<br>=><br>TRIG_WD | WAIT-CFG |
| 69 | WAIT-CFG | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | WAIT-CFG |
| 70 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE<br>&& Diag.Master_Add = Req_Add<br>=><br>Diag.Master_Add:= invalid<br>Diag.Prm_Req:= TRUE<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 71 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE<br>&& Diag.Master_Add <> Req_Add<br>=><br>ignore | WAIT-CFG |
| 72 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = FALSE && Lock_Req = TRUE<br>&& Diag.Master_Add = Req_Add && (Ident_Number <> Diag.Ident_Number<br>\|\| OPERATION_MODE_OK = FALSE \|\| WD_On = TRUE && (WD_Fact_1=0 \|\|<br>WD_Fact_2=0))<br>=><br>Diag.Prm_Fault:= TRUE<br>Diag.Master_Add:= Invalid<br>Diag.Prm_Req:= TRUE<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 73 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = FALSE && Lock_Req = TRUE<br>&& Diag.Master_Add = Req_Add && Ident_Number = Diag.Ident_Number<br>&& OPERATION_MODE_OK && (WD_On = FALSE \|\| WD_Fact_1>0 &&<br>WD_Fact_2>0)<br>&& (Freeze_Req = TRUE && Freeze Supp = FALSE<br>\|\| Sync_Req = TRUE && Sync Supp = FALSE \|\|<br>Prm_Data[1].0, .1, .2 = TRUE)<br>=><br>Diag.Not_Supported:= TRUE<br>Diag.Master_Add:= Invalid<br>Diag.Prm_Req:= TRUE<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 74 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = FALSE && Lock_Req = TRUE<br>&& Diag.Master_Add <> Req_Add && (Ident_Number <> Diag.Ident_Number<br>\|\| OPERATION_MODE_OK = FALSE \|\| WD_On = TRUE && (WD_Fact_1=0 \|\|<br>WD_Fact_2=0)<br>\|\| Freeze_Req = TRUE && Freeze Supp = FALSE<br>\|\| Sync_Req = TRUE && Sync Supp = FALSE<br>\|\| Prm_Data[1].0, .1, .2 = TRUE )<br>=><br>ignore | WAIT-CFG |
| 75 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/PRM_OK && Prm_Pending = 0<br>=><br>Diag.Master_Add:= Req_Add<br>Check_Prm_Add:= Req Add<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups:= Group_Ident<br>Prm_Pending:= 1<br>User Parameter Data:= Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Input_Pending:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 76 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/PRM_OK && Prm_Pending > 0<br>=><br>Diag.Master_Add:= Req_Add<br>Check_Prm_Add:= Req Add<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups:= Group_Ident<br>Prm_Pending:= 2<br>B User Prm:= Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Input_Pending:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | WAIT-CFG |
| 77 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = FALSE && Lock_Req = FALSE<br>=><br>DMPMS Set minTsdr.req ( MinTsdr ) | WAIT-CFG |
| 78 | WAIT-CFG | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len < 7<br>=><br>ignore | WAIT-CFG |
| 79 | WAIT-CFG | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | WAIT-CFG |
| 80 | WAIT-CFG | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add = Req_Add && Cfg_Pending = 0<br>=><br>Cfg_Pending:= 1<br>Check_Cfg_Add:= Req Add<br>TRIG_WD<br>MSCY1S Check Cfg.ind ( Check Cfg Mode, Cfg Data ) | WAIT-CFG |
| 81 | WAIT-CFG | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add = Req_Add && Cfg_Pending > 0<br>=><br>Cfg_Pending:= 2<br>Check_Cfg_Add:= Req Add<br>B Cfg:= Cfg Data<br>TRIG_WD | WAIT-CFG |
| 82 | WAIT-CFG | WDTimer expired<br>=><br>Diag.Master_Add:= invalid<br>Diag.Prm_Req:= TRUE<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 83 | WAIT-CFG | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add = Req_Add &&Ext_Prm_Pending = 0<br>=><br>TRIG_WD<br>Ext User Parameter Data:= Ext User Prm Data<br>Ext_Prm_Pending:= 1<br>Check_Ext_Prm_Add:= Req_Add<br>MSCY1S Check Ext User Prm.ind ( Ext User Parameter Data ) | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 84 | WAIT-CFG | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add = Req_Add &&<br>Ext_Prm_Pending > 0<br>=><br>TRIG_WD<br>B Ext User Prm:= Ext User Prm Data<br>Ext_Prm_Pending:= 2<br>Check_Ext_Prm_Add:= Req_Add | WAIT-CFG |
| 85 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | WAIT-CFG |
| 86 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 2<br>=><br>Status:= New Prm<br>Ext_Prm_Pending:= 1<br>Ext User Parameter Data:= B Ext User Prm<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status )<br>MSCY1S Check Ext User Prm.ind ( Ext User Parameter Data ) | WAIT-CFG |
| 87 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add <> Check_Ext_Prm_Add<br>=><br>Status:= Inv Master Add<br>Ext_Prm_Pending:= 0<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | WAIT-CFG |
| 88 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add = Check_Ext_Prm_Add &&<br>Ext_Prm_OK = TRUE<br>=><br>Ext_Prm_Pending:= 0<br>MSCY1S Check Ext User Prm Result.cnf(+)() | WAIT-CFG |
| 89 | WAIT-CFG | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Ext_Prm_OK = FALSE<br>=><br>Ext_Prm_Pending:= 0<br>Diag.Cfg_Fault:= TRUE<br>Diag.Master_Add:= Invalid<br>Diag.Prm_Req:= TRUE<br>StopTimer(WD)<br>Diag.WD_On:= FALSE<br>STOP_C1<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>MSCY1S Check Ext User Prm Result.cnf(+)()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 90 | DATA-EXCH | MSCY1S Abort.req ()<br>=><br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 91 | DATA-EXCH | MSCY1S Set Slave Diag.req ( Ext Diag Flag, Ext Diag Overflow, Ext Diag Data, Reference )<br>=><br>Diag.Ext Diag Data:= Ext Diag Data<br>Diag.Ext Diag Flag:= Ext Diag Flag<br>Diag.Ext Diag Overflow:= Ext Diag Overflow<br>Diag Data:= Diag<br>Diag Flag:= TRUE<br>Act_Ref:= Act_Ref + 1<br>Act_Cnt:= Reference<br>Ref_Cnt:= Act_Ref<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Data Exchange Upd.req( Diag Flag, Inp Data ) | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 92 | DATA-EXCH | MSCY1S Set Cfg.req ( Cfg Data )<br>=><br>Diag.Cfg_Fault:=TRUE<br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Get Cfg Upd.req ( Cfg Data )<br>MSCY1S Set Cfg.cnf(+) ()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 93 | DATA-EXCH | MSCY1S Get Output.req ()<br>/DX_Entered = TRUE<br>=><br>Clear Flag:= Safe_State<br>New Flag:= New Output<br>New Output:= FALSE<br>Outp Data, Output Data:= B Output<br>DMPMS RD Outp Upd.req ( Outp Data )<br>MSCY1S Get Output.cnf(+) ( Output Data, Clear Flag, New Flag ) | DATA-EXCH |
| 94 | DATA-EXCH | MSCY1S Get Output.req ()<br>/DX_Entered = FALSE<br>=><br>MSCY1S Get Output.cnf(-) () | DATA-EXCH |
| 95 | DATA-EXCH | MSCY1S Set Input.req ( Input Data )<br>/Diag.Freeze_Mode = FALSE<br>=><br>Inp Data, B Input:= Input Data<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>MSCY1S Set Input.cnf(+) () | DATA-EXCH |
| 96 | DATA-EXCH | MSCY1S Set Input.req ( Input Data )<br>/Diag.Freeze_Mode = TRUE<br>=><br>B Freeze:= Input Data<br>MSCY1S Set Input.cnf(+) () | DATA-EXCH |
| 97 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | DATA-EXCH |
| 98 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 2<br>=><br>Status:= New Prm<br>Prm_Pending:= 1<br>User Parameter Data:= B User Prm<br>MSCY1S Check User Prm Result.cnf(-) ( Status )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | DATA-EXCH |
| 99 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add # Check_Prm_Add<br>=><br>Status:= Inv Master Add<br>Prm_Pending:= 0<br>MSCY1S Check User Prm Result.cnf(-) ( Status ) | DATA-EXCH |
| 100 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Prm_OK = TRUE<br>=><br>Prm_Pending:= 0<br>MSCY1S Check User Prm Result.cnf(+)() | DATA-EXCH |
| 101 | DATA-EXCH | MSCY1S Check User Prm Result.req ( Prm_OK )<br>/Prm_Pending = 1 && Diag.Master_Add = Check_Prm_Add && Prm_OK = FALSE<br>=><br>Diag.Prm_Fault:= TRUE<br>LEAVE-MASTER<br>Prm_Pending:= 0<br>Act_Ref:= Act_Ref + 1<br>MSCY1S Check User Prm Result.cnf(+)()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 102 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | DATA-EXCH |
| 103 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 2<br>=><br>Status:= New Cfg<br>Cfg_Pending:= 1<br>Cfg Data:= B Cfg<br>MSCY1S Check Cfg Result.cnf(-) ( Status )<br>MSCY1S Check Cfg.ind ( Cfg Data ) | DATA-EXCH |
| 104 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add # Check_Cfg_Add<br>=><br>Status:= Inv Master Add<br>Cfg_Pending:= 0<br>MSCY1S Check Cfg Result.cnf(-) ( Status ) | DATA-EXCH |
| 105 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Input Data Len = Inp Data Len<br>&& Output Data Len = Outp Data Len<br>=><br>Cfg_Pending:= 0<br>MSCY1S Check Cfg Result.cnf(+) () | DATA-EXCH |
| 106 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = TRUE<br>&& Input Data Len <> Inp Data Len<br>&& Output Data Len <> Outp Data Len<br>=><br>Diag.Cfg_Fault:= TRUE<br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>MSCY1S Check Cfg Result.cnf(+) ()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 107 | DATA-EXCH | MSCY1S Check Cfg Result.req ( Cfg_OK, Input Data Len, Output Data Len )<br>/Cfg_Pending = 1 && Diag.Master_Add = Check_Cfg_Add && Cfg_OK = FALSE<br>=><br>Diag.Cfg_Fault:= TRUE<br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>MSCY1S Check Cfg Result.cnf(+) ()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 108 | DATA-EXCH | MSCY1S Application Ready.req()<br>/Diag.Stat_Diag = TRUE<br>=><br>Diag.Stat_Diag:= FALSE<br>Diag_Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Data Exchange Upd.req( Diag Flag, Inp Data ) | DATA-EXCH |
| 109 | DATA-EXCH | MSCY1S Application Ready.req()<br>/Diag.Stat_Diag = FALSE<br>=><br>ignore | DATA-EXCH |
| 110 | DATA-EXCH | MSAC1S Abort.ind()<br>=><br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 111 | DATA-EXCH | MSAC1S Start.cnf()<br>/Start_State <> B<br>=><br>START_C1_CON | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 112 | DATA-EXCH | MSAC1S Start.cnf()<br>/Start_State = B<br>=><br>START_C1_CON<br>MSCY1S_Start.ind ( Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit) | DATA-EXCH |
| 113 | DATA-EXCH | MSAC1S Stop.cnf()<br>/Start_State <> E<br>=><br>STOP_C1_CON | DATA-EXCH |
| 114 | DATA-EXCH | MSAC1S Stop.cnf()<br>/Start_State = E<br>=><br>STOP_C1_CON<br>MSCY1S_Stop.ind | DATA-EXCH |
| 115 | DATA-EXCH | DMPMS Set Slave Add.ind(New Slave Add, Ident Number, No Add Chg, Rem Slave Data)<br>=><br>ignore | DATA-EXCH |
| 116 | DATA-EXCH | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add = Req_Add &&<br>( Diag.Stat_Diag = TRUE \|\| Reference < Ref_Cnt)<br>=><br>TRIG_WD | DATA-EXCH |
| 117 | DATA-EXCH | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add = Req_Add && Diag.Stat_Diag = FALSE && Reference >= Ref_Cnt<br>=><br>Diag Flag:= FALSE<br>TRIG_WD<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>MSCY1S Set Slave Diag.cnf(+) (Reference:= Act_Cnt ) | DATA-EXCH |
| 118 | DATA-EXCH | DMPMS Slave Diag.ind(Req Add, Reference)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | DATA-EXCH |
| 119 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = FALSE && Unlock_Req = FALSE<br>=><br>DMPMS Set minTsdr.req ( MinTsdr ) | DATA-EXCH |
| 120 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE && Diag.Master_Add <> Req_Add<br>=><br>ignore | DATA-EXCH |
| 121 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Unlock_Req = TRUE && Diag.Master_Add = Req_Add<br>=><br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 122 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE<br>&& Diag.Master_Add = Req_Add && (Ident_Number <> Diag.Ident_Number<br>\|\| !OPERATION_MODE_OK \|\| WD_On && (WD_Fact_1=0 \|\| WD_Fact_2=0))<br>=><br>Diag.Prm_Fault:= TRUE<br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 123 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE<br>&& Diag.Master_Add = Req_Add && Ident_Number = Diag.Ident_Number<br>&& OPERATION_MODE_OK && !WD_On \|\| (WD_Fact_1>0 && WD_Fact_2>0)<br>&& (Freeze_Req && Freeze Supp = FALSE<br>\|\| Sync_Req && Sync Supp = FALSE<br>\|\| Prm_Data[1].0, .1, .2 = TRUE)<br>=><br>Diag.Not_Supported:= TRUE<br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 124 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len >= 7 && Lock_Req = TRUE && Unlock_Req = FALSE<br>&& Diag.Master_Add <> Req_Add && (Ident_Number <> Diag.Ident_Number<br>\|\| !OPERATION_MODE_OK \|\| WD_On && (WD_Fact_1=0 \|\| WD_Fact_2=0)<br>\|\| Freeze_Req && !Freeze Supp<br>\|\| Sync_Req && !Sync Supp<br>\|\| Prm_Data[1].0, .1, .2 = TRUE)<br>=><br>ignore | DATA-EXCH |
| 125 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK &&!PV0V0 && NOPRMCMD<br>=><br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 126 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK &&<br>!NOPRMCMD && Prm_Pending = 0<br>=><br>SET_WD<br>Active_Groups:= Group_Ident<br>EXE_PRM_CMD<br>Prm_Pending:= 1<br>Check_Prm_Add:= Req Add<br>User Parameter Data:= Prm Data.User Prm<br>Diag_Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | DATA-EXCH |
| 127 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK &&<br>!NOPRMCMD && Prm_Pending > 0<br>=><br>SET_WD<br>Active_Groups:= Group_Ident<br>EXE_PRM_CMD<br>Prm_Pending:= 2<br>Check_Prm_Add:= Req Add<br>B User Prm:= Prm Data.User Prm<br>Diag_Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 128 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK && NOPRMCMD && PV0V0 &&<br>Prm_Pending = 0<br>=><br>SET_WD<br>Active_Groups:= Group_Ident<br>Prm_Pending:= 1<br>Check_Prm_Add:= Req Add<br>User Parameter Data:= Prm Data.User Prm<br>Diag_Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | DATA-EXCH |
| 129 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add = Req_Add && PRM_OK && NOPRMCMD && PV0V0 &&<br>Prm_Pending > 0<br>=>SET_WD<br>Active_Groups:= Group_Iden<br>tPrm_Pending:= 2<br>Check_Prm_Add:= Req Add<br>B User Prm:= Prm Data.User Prm<br>Diag_Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | DATA-EXCH |
| 130 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add <> Req_Add && PRM_OK && Prm_Pending = 0<br>=><br>Diag.Sync_Mode:= FALSE<br>Diag.Freeze_Mode:= FALSE<br>Diag.Station_Not_Ready:= TRUE<br>Diag.Stat_Diag:= FALSE<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups:= Group_Ident<br>Safe_State:= TRUE<br>Diag.Master_Add:= Req_Add<br>Check_Prm_Add:= Req Add<br>STOP_ASM<br>STOP_C1<br>Prm_Pending:= 1<br>User Parameter Data:= Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Leave.req<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr )<br>MSCY1S Check User Prm.ind ( Prm Structure, User Parameter Data ) | WAIT-CFG |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 131 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Diag.Master_Add <> Req_Add && PRM_OK && Prm_Pending > 0<br>=><br>Diag.Sync_Mode:= FALSE<br>Diag.Freeze_Mode:= FALSE<br>Diag.Station_Not_Ready:= TRUE<br>Diag.Stat_Diag:= FALSE<br>SET_OPERATION_MODE<br>SET_WD<br>Active_Groups:= Group_Ident<br>Safe_State:= TRUE<br>Diag.Master_Add:= Req_Add<br>Check_Prm_Add:= Req Add<br>STOP_ASM<br>STOP_C1<br>Prm_Pending:= 2<br>B User Prm:= Prm Data.User Prm<br>SET_ALARM_CHKCFGM<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Leave.req<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>DMPMS Set minTsdr.req ( minTsdr ) | WAIT-CFG |
| 132 | DATA-EXCH | DMPMS Set Prm.ind(Req Add, Prm Data)<br>/Prm_Data.len < 7<br>=><br>ignore | DATA-EXCH |
| 133 | DATA-EXCH | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | DATA-EXCH |
| 134 | DATA-EXCH | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add = Req_Add && Cfg_Pending = 0<br>=><br>Cfg_Pending:= 1<br>Check_Cfg_Add:= Req Add<br>TRIG_WD<br>MSCY1S Check Cfg.ind ( Check Cfg Mode, Cfg Data ) | DATA-EXCH |
| 135 | DATA-EXCH | DMPMS Chk Cfg.ind(Req Add, Cfg Data)<br>/Diag.Master_Add = Req_Add && Cfg_Pending > 0<br>=><br>Cfg_Pending:= 2<br>Check_Cfg_Add:= Req Add<br>B Cfg:= Cfg Data<br>TRIG_WD | DATA-EXCH |
| 136 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len > 0 && Outp_Data.len = Outp Data Len<br>&& Diag.Sync_Mode = FALSE<br>=><br>B Output:= Outp Data<br>TRIG_WD<br>Safe_State:= FALSE<br>Clear Flag:= FALSE<br>New Output:= TRUE<br>CHECK_ALARM_START<br>MSCY1S New Output.ind ( Clear Flag ) | DATA-EXCH |
| 137 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len > 0 && Outp_Data.len = Outp Data Len<br>&& Diag.Sync_Mode = TRUE<br>=><br>B Sync:= Outp Data<br>TRIG_WD<br>Safe_State:= FALSE<br>CHECK_ALARM_START | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 138 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len = 0 && Outp Data Len = 0<br>=><br>TRIG_WD<br>CHECK_ALARM_START | DATA-EXCH |
| 139 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len <> Outp Data Len && (Outp_Data.len = 0 && Fail_Safe_supp)<br>=><br>TRIG_WD<br>Safe_State:= TRUE<br>Clear Flag:= TRUE<br>New Output:= TRUE<br>CHECK_ALARM_START<br>MSCY1S New Output.ind ( Clear Flag ) | DATA-EXCH |
| 140 | DATA-EXCH | DMPMS Data Exchange.ind(Outp Data)<br>/Outp_Data.len <> Outp Data Len &&<br>(Outp_Data.len <> 0 \|\| Fail_Safe_supp )<br>=><br>LEAVE-MASTERAct_Ref:= Act_Ref + 1<br>Diag.Cfg_Fault:= TRUE<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 141 | DATA-EXCH | WDTimer expired<br>=><br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 142 | DATA-EXCH | DMPMS Global Control.ind(Control Command, Group Select)<br>/ (Control_Command.0, .6, .7 = TRUE)<br>=><br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 143 | DATA-EXCH | DMPMS Global Control.ind(Control Command, Group Select)<br>/Group_Select <> 0 && ((Active_Groups & Group_Select) = 0) && ((Group_Select < 0x80 ) \|\| Isochronous Mode = FALSE) && (Control_Command.0, .6, .7 = FALSE)<br>=><br>ignore | DATA-EXCH |
| 144 | DATA-EXCH | DMPMS Global Control.ind(Control Command, Group Select)<br>/Control Command.Clear Command = FALSE<br>&& (Group_Select = 0 \|\| (Active_Groups & Group_Select) <> 0<br>\|\| ((Group_Select >= 0x80 ) && Isochronous Mode = TRUE))<br>&& (Control_Command.0, .6, .7 = FALSE)<br>=><br>Clear Command:= FAUX | CHECK-ISOM |
| 145 | DATA-EXCH | DMPMS Global Control.ind(Control Command, Group Select)<br>/Control Command.Clear Command = TRUE<br>&& (Group_Select = 0 \|\| (Active_Groups & Group_Select) <> 0<br>\|\| ((Group_Select >= 0x80 ) && Isochronous Mode = TRUE))<br>&& (Control_Command.0, .6, .7 = FALSE)<br>=><br>Clear Command:= TRUE<br>Clear Flag:= TRUE<br>New Output:= TRUE<br>Safe_State:= VRAI | CHECK-ISOM |
| 146 | DATA-EXCH | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add = Req_Add<br>=><br>Diag.Prm_Fault:= TRUE<br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 147 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 0<br>=><br>Status:= Not pending<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | DATA-EXCH |
| 148 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 2<br>=><br>Status:= New Prm<br>Ext_Prm_Pending:= 1<br>Ext_User Parameter Data:= B Ext User Prm<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status )<br>MSCY1S Check Ext User Prm.ind ( Ext User Parameter Data ) | DATA-EXCH |
| 149 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add <> Check_Ext_Prm_Add<br>=><br>Status:= Inv Master Add<br>Ext_Prm_Pending:= 0<br>MSCY1S Check Ext User Prm Result.cnf(-) ( Status ) | DATA-EXCH |
| 150 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add = Check_Ext_Prm_Add &&<br>Ext_Prm_OK = TRUE<br>=><br>Ext_Prm_Pending:= 0<br>MSCY1S Check Ext User Prm Result.cnf(+)() | DATA-EXCH |
| 151 | DATA-EXCH | MSCY1S Check Ext User Prm Result.req ( Ext_Prm_OK )<br>/Ext_Prm_Pending = 1 && Diag.Master_Add = Check_Ext_Prm_Add &&<br>Ext_Prm_OK = FALSE<br>=><br>Diag.Cfg_Fault:= TRUE<br>LEAVE-MASTER<br>Ext_Prm_Pending:= 0<br>Act_Ref:= Act_Ref + 1<br>MSCY1S Check Ext User Prm Result.cnf(+)()<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 152 | CHECK-SYNC | /NOSYNC<br>=><br>Sync Command:= 0 | CHECK-FREEZE |
| 153 | CHECK-SYNC | /UNSYNC && Sync Supp = FALSE<br>=><br>Sync Command:= 2 | CHECK-FREEZE |
| 154 | CHECK-SYNC | /SYNC && Sync Supp = FALSE<br>=><br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 155 | CHECK-SYNC | /UNSYNC && Sync Supp = TRUE && B Sync <> Nil<br>=><br>B Output:= B Sync<br>B Sync:= Nil<br>Sync Command:= 2<br>Clear Flag:= Safe_State<br>New Outpu t:= TRUE<br>Diag.Sync_Mode:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>MSCY1S New Output.ind ( Clear Flag )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | CHECK-FREEZE |
| 156 | CHECK-SYNC | /UNSYNC && Sync Supp = TRUE && B Sync = Nil<br>=><br>Sync Command:= 2<br>Clear Flag:= Safe_State<br>Diag.Sync_Mode:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | CHECK-FREEZE |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 157 | CHECK-SYNC | /SYNC && Sync Supp = TRUE && Diag.Sync_Mode = FALSE<br>=><br>B Sync:= Nil<br>Sync Command:= 1<br>Diag.Sync_Mode:= TRUE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | CHECK-FREEZE |
| 158 | CHECK-SYNC | /SYNC && Sync Supp = TRUE && Diag.Sync_Mode = TRUE && B Sync <> Nil<br>=><br>B Output:= B Sync<br>B Sync:= Nil<br>Clear Flag:= Safe State<br>New Output:= TRUE<br>Sync Command:= 1<br>MSCY1S New Output.ind ( Clear Flag ) | CHECK-FREEZE |
| 159 | CHECK-SYNC | /SYNC && Sync Supp = TRUE && Diag.Sync_Mode = TRUE && B Sync = Nil<br>=><br>Sync Command:= 1 | CHECK-FREEZE |
| 160 | CHECK-ISOM | /(Group_Select >= 0x80 ) && Isochronous Mode = TRUE<br>=><br>if (FirstSynch = TRUE)<br>Status:= IsoM Start<br>FirstSynch:= FALSE<br>else<br>Status:=IsoM SYNCH<br>endif<br>MSCY1S SYNCH Event.ind(Status) | CHECK-SYNC |
| 161 | CHECK-ISOM | /(Group_Select < 0x80 ) \|\| Isochronous Mode = FALSE<br>=> | CHECK-SYNC |
| 162 | CHECK-FREEZE | /NOFREEZE<br>=><br>Freeze Command:= 0<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 163 | CHECK-FREEZE | /UNFREEZE && Freeze Supp = FALSE<br>=><br>Freeze Command:= 2<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 164 | CHECK-FREEZE | /FREEZE && Freeze Supp = FALSE<br>=><br>LEAVE-MASTER<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref ) | WAIT-PRM |
| 165 | CHECK-FREEZE | /UNFREEZE && Freeze Supp = TRUE && B Freeze <> Nil<br>=><br>Inp Data:= B Freeze<br>B Freeze:= Nil<br>Freeze Command:= 2<br>Diag.Freeze_Mode:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /Condition =>Action | Prochain état |
|---|---|---|---|
| 166 | CHECK-FREEZE | /UNFREEZE && Freeze Supp = TRUE && B Freeze = Nil<br>=><br>Freeze Command:= 2<br>Diag.Freeze_Mode:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 167 | CHECK-FREEZE | /FREEZE && Freeze Supp = TRUE && Diag.Freeze_Mode = FALSE<br>=><br>B Freeze:= Nil<br>Freeze Command:= 1<br>Diag.Freeze_Mode:= FALSE<br>Diag Data:= Diag<br>Act_Ref:= Act_Ref + 1<br>DMPMS Slave Diag Upd.req ( Diag Data, Reference:= Act_Ref )<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 168 | CHECK-FREEZE | /FREEZE && Freeze Supp = TRUE && Diag.Freeze_Mode = TRUE && B Freeze <> Nil<br>=><br>Inp Data:= B Freeze<br>B Freeze:= Nil<br>Freeze Command:= 1<br>DMPMS Data Exchange Upd.req ( Diag Flag, Inp Data )<br>DMPMS RD Inp Upd.req ( Inp Data )<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 169 | CHECK-FREEZE | /FREEZE && Freeze Supp = TRUE && Diag.Freeze_Mode = TRUE && B Freeze = Nil<br>=><br>Freeze Command:= 1<br>MSCY1S Global Control.ind ( Clear Command, Sync Command, Freeze Command ) | DATA-EXCH |
| 170 | tout état | MSCY1S Reset.req ()<br>=><br>DMPMS Slave Deact.req () | SL-DEACT |
| 171 | tout état | DMPMS Set Ext Prm.ind(Req_Add, Ext Prm Data)<br>/Diag.Master_Add <> Req_Add<br>=><br>ignore | SAME |
| 172 | SL-DEACT | DMPMS Slave Deact.cnf ()<br>=><br>MSCY1S Reset.cnf () | |

### 9.1.4 Fonctions

Le Tableau 49 contient les fonctions utilisées par le MSCY1S et leurs descriptions.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

**Tableau 49 – Fonctions utilisées par le MSCY1S**

| Nom | Fonction |
|---|---|
| LEAVE-MASTER | Diag.Master_Add=invalid<br>Diag.Sync_Mode=FALSE<br>Diag.Freeze_Mode=FALSE<br>Diag.Prm_Req=TRUE<br>Diag.Station_Not_Ready=TRUE<br>Stop Timer(WD), Diag.WD_On=FALSE<br>Diag.Stat_Diag=FALSE, Operation_Mode=V0<br>Safe_State=TRUE<br>STOP_ASM,<br>STOP_C1,<br>STOP_ISOM<br>DMPMS_Leave.Req<br>Diag_Data=Diag |
| PRM_OK | Prm_Data.len>=7<br>&& Lock_Req=TRUE<br>&& Unlock_Req=FALSE<br>&& Ident_Number=Diag.Ident_Number<br>&& (WD_On=FALSE ||<br>(WD_On=TRUE && WD_Fact_1>0 && WD_Fact_2>0))<br>&& (Freeze_Req=FALSE || Freeze_Supported)<br>&& (Sync_Req=FALSE || Sync_Supported)<br>&& (Prm_Data[1].0, .1, .2=FALSE)<br>&& OPERATION_MODE_OK |
| SET_WD | if (WD_On=TRUE)<br>Start Timer(WD),<br>Diag.WD_On=TRUE)<br>else<br>Stop Timer(WD),<br>Diag.WD_On=FALSE<br>endif |
| TRIG_WD | if (Diag.WD_On=TRUE)<br>Start Timer(WD)<br>endif |
| NOSYNC | Control_Command.Sync=0 && Control_Command.UnSync=0 |
| SYNC | Control_Command.Sync=1 && Control_Command.UnSync=0 |
| UNSYNC | Control_Command.UnSync=1 |
| NOFREEZE | Control_Command.Freeze=0 && Control_Command.UnFreeze=0 |
| FREEZE | Control_Command.Freeze=1 && Control_Command.UnFreeze=0 |
| UNFREEZE | Control_Command.UnFreeze=1 |
| STOP_ISOM | if(Isochronous Mode Supp = TRUE<br>&& Isochronous Mode =TRUE<br>&& First Synch = FALSE)<br>MSCY1S SYNCH Event.ind(Status:= IsoM Stop)<br>First Synch = TRUE<br>endif |
| STOP_ASM | if (DX_Entered=TRUE)<br>MSCY1S_Stop.ind<br>endif |
| CHECK_ALARM_START | if (DX_Entered=FALSE && Diag.Stat_Diag=FALSE)<br>DX_Entered=TRUE,<br>MSCY1S_Start.ind(Actual_Enabled_Alarms, Alarm_Sequence, Alarm_Limit)<br>endif |
| SET_OPERATION_MODE | if (Prm_Data.len >= 10 && DPV1_Enable=TRUE)<br>Operation_Mode=V1<br>else<br>Operation_Mode=V0<br>endif<br>if (Prm_Data.len >= 10 && IsoM_Req =TRUE)<br>Isochron_Mode=True<br>else<br>Isochron_Mode=FALSE<br>endif |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom | Fonction |
|---|---|
| PV0V0 | Operation_Mode = V0<br>&& (Prm_Data.len < 10 \|\| DPV1_Enable = FALSE) |
| OPERATION_MODE_OK | if (IsARexistent(MS1) && DPV1_Enable )<br>further checks according the table “Rules for DPV1_Status_1, DPV1_Status_2, and DPV1_Status_3 check” |
| NOPRMCMD | !(DPV1_Status_3.PrmCmd = 1 && Prm_Data.len = 18 && PrmCmd_supp) |
| SET_ALARM_CHKCFGM | if (Prm_Data.len >=10 && DPV1_Enable=TRUE)<br>Enabled_Alarms=DPV1_Status_2.2 - DPV1_Status_2.7<br>else<br>Enabled_Alarms=0<br>endif<br>if (Prm_Data.len >=10)<br>Alarm_Mode_Master=DPV1_Status_3.0 - DPV1_Status_3.2,<br>Chk Cfg Mode = DPV1_Status_2.0,<br>Prm Structure = DPV1_Status_3.4,<br>if (DPV1_Status3.PrmCmd&& Prm_Data.PrmCmd exist)<br>EXE_PRM_CMD<br>else<br>Alarm_Mode_Master=0,<br>Chk Cfg Mode = FALSE,<br>Prm Structure = FALSE<br>endif |
| EXE_PRM_CMD | if (PrmCmd.Function.Primary)<br>Primary=TRUE,<br>if(Start_State=B\|\|Start_State=BEB\|\|Start_State=EB\|\|Start_State=Run)<br>STOP_C1, START_C1<br>else<br>START_C1<br>endif<br>else<br>Primary=FALSE<br>endif<br>if (PrmCmd.Function.Stop_MS1)<br>if(Start_State=B\|\|Start_State=BEB\|\|Start_State=EB\|\|Start_State=Run)<br>STOP_C1<br>endif<br>endif<br>if (PrmCmd.Function.Start_MS1)<br>if(Start_State=E\|\|Start_State=BE\|\|Start_State=Idle)<br>START_C1<br>endif<br>endif |
| START_C1 | if (Start_State=Idle)<br>MSAC1S_Start.req(Diag.Master_Add),<br>Start_State=B<br>endif<br>if (Start_State=E)<br>Start_State=EB<br>endif<br>if (Start_State=BE)<br>Start_State=BEB<br>endif |
| STOP_C1 | if (Start_State=B)<br>Start_State=BE<br>endif<br>if (Start_State=Run)<br>MSAC1S_Stop.req,<br>Start_State=E<br>endif<br>if (Start_State=EB)<br>Start_State=E<br>endif<br>if (Start_State=BEB)<br>Start_State=BE<br>endif |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| Nom | Fonction |
|---|---|
| START_C1_CON | if (Start_State=B)<br>  Start_State=Run<br>endif<br>if (Start_State=BE)<br>  MSAC1S_Stop.req, Start_State=E<br>endif<br>if (Start_State=BEB)<br>  MSAC1S_Stop.req, Start_State=EB<br>endif |
| STOP_C1_CON | if (Start_State=E)<br>  Start_State=Idle<br>endif<br>if (Start_State=EB)<br>  MSAC1S_Start.req(Diag.Master_Add),<br>  Start_State=B<br>endif |
| LEAVE-MASTER | Diag.Master_Add=invalid<br>Diag.Sync_Mode=FALSE<br>Diag.Freeze_Mode=FALSE<br>Diag.Prm_Req=TRUE<br>Diag.Station_Not_Ready=TRUE<br>Stop Timer(WD), Diag.WD_On=FALSE<br>Diag.Stat_Diag=FALSE, Operation_Mode=V0<br>Safe_State=TRUE<br>STOP_ASM,<br>STOP_C1,<br>STOP_ISOM<br>DMPMS_Leave.Req<br>Diag_Data=Diag |

## 9.2 MSAC1S

### 9.2.1 Définitions des primitives

#### 9.2.1.1 Primitives échangées entre FSPMS et MSAC1S

Le Tableau 50 montre les primitives de service, y compris leurs paramètres associés, émises par le FSPMS et reçues par le MSAC1S.

**Tableau 50 – Primitives émises par FSPMS vers MSAC1S**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| SInit MS1.req | FSPMS | (aucun) | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3. |
| Reset.req | FSPMS | (aucun) | |
| Read.rsp(+) | FSPMS | Length,<br>Data | |
| Read.rsp(-) | FSPMS | Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Write.rsp(+) | FSPMS | Longueur | |
| Write.rsp(-) | FSPMS | Error Decode,<br>Error Code 1<br>Error Code 2 | |
| Alarm Ack.rsp(+) | FSPMS | Slot Number,<br>Alarm Type,<br>Seq Nr | |
| Alarm Ack.rsp(-) | FSPMS | (aucun) | |

Le Tableau 51 montre les primitives de service, y compris leurs paramètres associés, émises par le MSAC1S et reçues par le FSPMS.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**Tableau 51 – Primitives émises par MSAC1S vers FSPMS**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| SInit MS1.cnf | MSAC1S | (aucun) | Se référer à la Définition des services de la FAL dans la CEI 61158-5-3. |
| Reset.cnf | MSAC1S | (aucun) | |
| Fault.ind | MSAC1S | (aucun) | |
| Read.ind | MSAC1S | Slot Number,<br>Index,<br>Length | |
| Write.ind | MSAC1S | Slot Number,<br>Index,<br>Length,<br>Data | |
| Alarm Ack.ind | MSAC1S | Slot Number,<br>Alarm Type,<br>Seq Nr | |

#### 9.2.1.2 Primitives échangées entre MSAC1S et MSCY1S

Le Tableau 52 montre les primitives de service, y compris leurs paramètres associés, émises par le MSCY1S et reçues par le MSAC1S.

**Tableau 52 – Primitives émises par MSCY1S vers MSAC1S**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Start.req | MSCY1S | Master Add | Démarrer le traitement des services acycliques |
| Stop.req | MSCY1S | (aucun) | Arrêter le traitement des services acycliques |

Le Tableau 53 montre les primitives de service, y compris leurs paramètres associés, émises par le MSAC1S et reçues par le MSCY1S.

**Tableau 53 – Primitives émises par MSAC1S vers MSCY1S**

| Nom de primitive | Source | Paramètres associés | Fonctions |
|---|---|---|---|
| Start.cnf | MSAC1S | (aucun) | MSAC1 est prêt à exécuter des services acycliques. |
| Stop.cnf | MSAC1S | (aucun) | MSAC1 est désactivé. |
| Abort.ind | MSAC1S | (aucun) | Arrêt de MSAC1 en raison d'une erreur de séquence ou en raison d'éléments de protocole erronés |

#### 9.2.1.3 Paramètres des primitives de MSAC1S

Les paramètres utilisés avec les primitives échangées entre le FSPMS et le MSAC1S sont décrits dans "Définition des services de la FAL" (voir la CEI 61158-5-3). Des paramètres supplémentaires sont décrits dans le Tableau 54.

**Tableau 54 – Paramètre utilisé avec les primitives échangées entre le diagramme MSAC1S et MSCY1S**

| Nom de paramètre | Description |
|---|---|
| MasterAdd | Ce paramètre achemine l'adresse DL du maître DP assigné. |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

### 9.2.2 Description de diagramme d'états

Après mise sous tension, le MSAC1S attend le lancement au moyen de SInit MS1.

Dans l'état CLOSED, le service Start appelé depuis le MSCY1S est attendu. Puis la protection d'accès pour le SAP 51 (50) sera activée au moyen de la fonction RSAP_ACTIVATE.

Lors de la réception de la confirmation, le diagramme d'états entre dans l'état OPEN. Dans cet état, les services Read, Write et Alarm_Ack sont autorisés. Une seule demande provenant du maître DP est traitée simultanément sur chaque SAP. En plus de Read/Write au niveau du SAP 51, un Alarm_Ack peut être exécuté au SAP 50.

Lorsqu'un service valide est invoqué par le maître DP, le MSAC1S indiquera cette primitive à l'AP-Context (Read/Write ou Alarm_Ack) et attend la réponse. Après réception de la réponse correspondante, le MSAC1S fournira cette PDU avec le tampon de mises à jour de SAP correspondant. Le service est fini après que le maître DP a récupéré la réponse.

Le service local Stop appelé depuis le MSCY1S ferme la connexion. Le SAP 51 (50) est désactivé et une réponse éventuellement stockée dans la DL est supprimée.

L'abandon de la connexion est indiqué au MSCY1S avec le service local Abort. Si l'utilisateur est toujours en train de traiter la réponse alors que la connexion est abandonnée, cette réponse doit être rejetée par l'utilisateur.

**Variables locales du MSAC1S**

**Server_SAP**
(Unsigned8)

Ce SAP (51) est utilisé pour les services MSAC1S_Read/_Write et MSAC1S_Alarm_Ack.

**Alarm_SAP**
(Unsigned8)

Ce SAP (50) est réservé pour les services MSAC1S_Alarm_Ack.

**Res_SAP**
(Unsigned8)

Variable contenant le SAP pour la réponse MSAC1S_Alarm_Ack.

**Service_Header**
(Unsigned8)

L'en-tête de service réel du service traité. Utilisé pour vérifier si l'utilisateur génère la réponse correcte.

**AA_State**
Les services MSAC1S_Alarm_Ack peuvent être exécutés simultanément avec les services MSAC1S_Read / MSAC1S_Write. Ainsi, l'état réel du traitement de MSAC1S_Alarm_Ack est stocké dans la variable AA_State:

Valeurs de l'AA_State:

Idle Aucun MSAC1S_Alarm_Ack en cours

WRes MSAC1S_Alarm_Ack.ind est envoyée au MSAL1S attendant une réponse.

Upd La réponse du MSAL1S sera envoyé au tampon de mises à jour (Update-Buffer).

SRes La réponse est stockée – dans l'attente du prochain sondage du maître DP.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

**VS_Upd**
Indique si une réponse sera envoyée à l'Update-Buffer.

### 9.2.3 Table d'états de MSAC1S

Le Tableau 55 contient la description complète du diagramme d'états MSCY1S.

**Tableau 55 Table d'états de MSAC1S**

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 1 | POWER-ON | MSAC1S_SInit_MS1.req<br>=><br>Server_SAP:=51<br>Alarm_SAP:=50<br>MSAC1S_SInit_MS1.cnf | CLOSED |
| 2 | POWER-ON | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 3 | POWER-ON | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 4 | POWER-ON | MSAC1S_Alarm_Ack.rsp(-) ()<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 5 | CLOSED | MSAC1S_Start.req(Master_Add)<br>=><br>DMPMS_RSAP_ACTIVATE.req(SSAP=Server_SAP, Access=Master_Add, L_sdu_length_list=Nil,Indication_Mode=Unchanged) | SET-ACC51 |
| 6 | CLOSED | MSAC1S_Stop.req<br>=><br>MSAC1S_Stop.cnf | CLOSED |
| 7 | SET-ACC51 | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status = OK<br>=><br>DMPMS_RSAP_ACTIVATE.req(SSAP=Alarm_SAP, Access=Master_Add, L_sdu_length_list=Nil,Indication_Mode=Unchanged) | SET-ACC50 |
| 8 | SET-ACC51 | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Server_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 9 | SET-ACC50 | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Alarm_SAP,M_status)<br>/M_status = OK<br>=><br>AA_State:=Idle<br>VS_Upd:=False<br>MSAC1S_Start.cnf | OPEN |
| 10 | SET-ACC50 | DMPMS_RSAP_ACTIVATE.cnf(SSAP=Alarm_SAP,M_status)<br>/M_status=NO/IV<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 11 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 12 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 13 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Read-REQ-PDU( )<br>=><br>Service_Header:=Function_Num<br>MSAC1S_Read.ind(Req_Add=Loc_add, Slot_Number,Index,Length) | VS-WRES |
| 14 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Write-REQ-PDU( ) && L_sdu.len >= (Length+4)<br>=><br>Service_Header:=Function_Num<br>MSAC1S_Write.ind(Req_Add=Loc_add, Slot_Number,Index,Length,Data) | VS-WRES |
| 15 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Alarm_Ack-REQ-PDU( )<br>=><br>AA_State:=WRes<br>Res_SAP:=Server_SAP<br>MSAC1S_Alarm_Ack.ind(Alarm_Type, Slot_Number,Seq_Nr) | AA-WRES |
| 16 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Alarm_Ack-REQ-PDU( )<br>=><br>AA_State:=WRes<br>Res_SAP:=Alarm_SAP<br>MSAC1S_Alarm_Ack.ind(Alarm_Type, Slot_Number,Seq_Nr) | AA-WRES |
| 17 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT(Read-REQ-PDU( ) \|\| Write-REQ-PDU( ) && (L_sdu.len >= Length+4) \|\| Alarm_Ack-REQ-PDU( ))<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 18 | OPEN | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT(Alarm_Ack-REQ-PDU( ))<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 19 | OPEN | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 20 | OPEN | MSAC1S_Read.rsp(+)(Length, Data)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 21 | OPEN | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 22 | OPEN | MSAC1S_Write.rsp(+)(Length)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 23 | OPEN | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 24 | OPEN | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 25 | OPEN | MSAC1S_Alarm_Ack.rsp(-) ()<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 26 | OPEN | MSAC1S_Stop.req<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | STOP-DEA51 |
| 27 | OPEN | MSAC1S_Start.req(Master_Add)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 28 | OPEN | MSAC1S_SInit_MS1.req<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 29 | OPEN | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 30 | VS-WRES | MSAC1S_Read.rsp(+)(Length, Data)<br>/Service_Header=0x5E && Max_DLSDU_Length_Req_Low>=(Length+4)<br>=><br>VS_Upd:=True<br>L_sdu:=Read-RES-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu,<br>Serv_class=Low,Transmit=Single) | VS-SRES |
| 31 | VS-WRES | MSAC1S_Read.rsp(+)(Length, Data)<br>/Service_Header=0x5E && Max_DLSDU_Length_Req_Low<(Length+4)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 32 | VS-WRES | MSAC1S_Write.rsp(+)(Length)<br>/Service_Header=0x5F<br>=><br>VS_Upd:=True<br>L_sdu:=Write-RES-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu,<br>Serv_class=Low,Transmit=Single) | VS-SRES |
| 33 | VS-WRES | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>/Service_Header=0x5E<br>=><br>VS_Upd:=True<br>L_sdu:=Read-NRS-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu,<br>Serv_class=Low,Transmit=Single) | VS-SRES |
| 34 | VS-WRES | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>/Service_Header=0x5F<br>=><br>VS_Upd:=True<br>L_sdu:=Write-NRS-PDU<br>DMPMS_REPLY_UPDATE.req(SSAP=Server_SAP, L_sdu,<br>Serv_class=Low,Transmit=Single) | VS-SRES |
| 35 | VS-WRES | MSAC1S_Read.rsp(+)(Length, Data)<br>/Service_Header<>0x5E<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 36 | VS-WRES | MSAC1S_Read.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>/Service_Header<>0x5E<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 37 | VS-WRES | MSAC1S_Write.rsp(+)(Length)<br>/Service_Header<>0x5F<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 38 | VS-WRES | MSAC1S_Write.rsp(-)(Error_Decode, Error_Code_1, Error_Code_2)<br>/Service_Header<>0x5F<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 39 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 40 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 41 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 42 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Alarm_Ack-REQ-PDU( ) && AA_State=Idle<br>=><br>AA_State:=WRes<br>Res_SAP:=Alarm_SAP<br>MSAC1S_Alarm_Ack.ind(Alarm_Type, Slot_Number,Seq_Nr) | VS-WRES |
| 43 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=SRes<br>=><br>AA_State:=Idle | VS-WRES |
| 44 | VS-WRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/NOT((Alarm_Ack-REQ-PDU( ) && AA_State=Idle)<br>\|\| (Update_status=LO && L_sdu.len=0 && AA_State=SRes))<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 45 | VS-WRES | MSAC1S_Start.req(Master_Add)<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 46 | VS-WRES | MSAC1S_Stop.req<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP) | STOP-DEA51 |
| 47 | VS-WRES | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>/AA_State<>WRes<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 48 | VS-WRES | MSAC1S_Alarm_Ack.rsp(-) ()<br>/AA_State<>WRes<br>=><br>DMPMS_SAP_DEACTIVATE.req(SSAP=Server_SAP)<br>MSAC1S_Abort.ind | ERR-DEA51 |
| 49 | VS-WRES | MSAC1S_Alarm_Ack.rsp(+) (Alarm_Type, Slot_Number, Seq_Nr)<br>/AA_State=WRes<br>=><br>AA_State:=Upd<br>DMPMS_REPLY_UPDATE.req(SSAP=Res_SAP, L_sdu=Alarm_Ack-RES-PDU, Serv_class=Low,Transmit=Single) | VS-WRES |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010
Withdrawn

| # | État courant | Événement /condition =>Action | Prochain état |
|---|---|---|---|
| 50 | VS-WRES | MSAC1S_Alarm_Ack.rsp(-) ()<br>/AA_State=WRes<br>=><br>AA_State:=Upd<br>DMPMS_REPLY_UPDATE.req(SSAP=Res_SAP, L_sdu=Alarm_Ack-NRS-PDU, Serv_class=Low,Transmit=Single) | VS-WRES |
| 51 | VS-WRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Alarm_SAP,Serv_class,L_status)<br>/L-status=OK && AA_State=Upd<br>=><br>AA_State:=SRes | VS-WRES |
| 52 | VS-WRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Alarm_SAP,Serv_class,L_status)<br>/L-status=OK && AA_State<>Upd<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 53 | VS-WRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L_status=LS/IV/LR<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 54 | VS-WRES | MSAC1S_SInit_MS1.req<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 55 | VS-WRES | MSAC1S_Reset.req<br>=><br>MSAC1S_Reset.cnf | POWER-ON |
| 56 | VS-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L-status=OK && VS_Upd=True<br>=><br>VS_Upd:=False | VS-SRES |
| 57 | VS-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L_status=LS/IV/LR<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 58 | VS-SRES | DMPMS_REPLY_UPDATE.cnf(SSAP=Server_SAP,Serv_class,L_status)<br>/L-status=OK && VS_Upd=False<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 59 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 60 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Alarm_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && Update_status=NO<br>=><br>MSAC1S_Fault.ind | POWER-ON |
| 61 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=Idle&& VS_Upd=False | OPEN |
| 62 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && AA_State=WRes && VS_Upd=False | AA-WRES |
| 63 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/Update_status=LO && L_sdu.len=0 && (AA_State=SRes\|\| AA_State=Upd) && VS_Upd=False | AA-SRES |
| 64 | VS-SRES | DMPMS_DATA_REPLY.ind(SSAP,DSAP=Server_SAP,Loc_add,Rem_add,L_sdu,Serv_class, Update_status)<br>/L_sdu.len=0 && VS_Upd=True<br>=><br>MSAC1S_Fault.ind | POWER-ON |

IECNORM.COM : Click to view the full PDF of IEC 61158-6-3:2010