

# INTERNATIONAL STANDARD

## NORME INTERNATIONALE



**Multimedia systems – Common communication protocol for inter-connectivity  
on heterogeneous networks**

**Systèmes multimédia – Protocole de communication commun relatif à la  
connectivité entre réseaux hétérogènes**

IECNORM.COM : Click to view the full PDF of IEC 62295:2007



## THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2007 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 Varembe  
CH-1211 Geneva 20  
Switzerland

Tel.: +41 22 919 02 11  
Fax: +41 22 919 03 00  
[info@iec.ch](mailto:info@iec.ch)  
[www.iec.ch](http://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](http://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It also gives information on projects, replaced and withdrawn publications.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](http://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](http://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](http://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](mailto: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](http://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](http://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](http://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](http://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](mailto:csc@iec.ch).

# INTERNATIONAL STANDARD

## NORME INTERNATIONALE



**Multimedia systems – Common communication protocol for inter-connectivity  
on heterogeneous networks**

**Systèmes multimédia – Protocole de communication commun relatif à la  
connectivité entre réseaux hétérogènes**

INTERNATIONAL  
ELECTROTECHNICAL  
COMMISSION

COMMISSION  
ELECTROTECHNIQUE  
INTERNATIONALE

PRICE CODE  
CODE PRIX

**XB**

ICS 33.040.40; 33.160; 35.100

ISBN 978-2-83220-244-9

**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.**

## CONTENTS

|                                                                                                    |    |
|----------------------------------------------------------------------------------------------------|----|
| FOREWORD.....                                                                                      | 6  |
| INTRODUCTION.....                                                                                  | 8  |
| 1 Scope and object.....                                                                            | 9  |
| 2 Normative references .....                                                                       | 10 |
| 3 Terms, definitions, abbreviations and conventions .....                                          | 10 |
| 3.1 Terms and definitions .....                                                                    | 10 |
| 3.2 Abbreviations .....                                                                            | 13 |
| 3.3 Conventions .....                                                                              | 14 |
| 4 Requirements .....                                                                               | 14 |
| 4.1 Home server interface requirements .....                                                       | 15 |
| 4.1.1 Basic requirements for data delivery .....                                                   | 15 |
| 4.1.2 Functional requirements for HNMP .....                                                       | 15 |
| 4.1.3 Home server interface requirements for unicast, multicast and<br>broadcast .....             | 15 |
| 4.2 CCP device requirements .....                                                                  | 16 |
| 4.2.1 Requirements for device registration .....                                                   | 16 |
| 4.2.2 Requirements for classification of CCP devices .....                                         | 16 |
| 5 Common communication protocol (CCP) layer .....                                                  | 17 |
| 5.1 CCP layer.....                                                                                 | 17 |
| 5.2 Data delivery over heterogeneous networks .....                                                | 19 |
| 6 CCP addressing .....                                                                             | 20 |
| 6.1 General.....                                                                                   | 20 |
| 6.2 An addressing structure to facilitate traffic switching for the home<br>server.Version 0 ..... | 20 |
| 6.2.1 Domain address .....                                                                         | 21 |
| 6.2.2 Cluster address .....                                                                        | 21 |
| 6.2.3 Device ID field .....                                                                        | 21 |
| 7 CCP packet format and fields .....                                                               | 21 |
| 7.1 General.....                                                                                   | 21 |
| 7.2 CCP packet format .....                                                                        | 22 |
| 7.2.1 CCP identification (CCPID).....                                                              | 22 |
| 7.2.2 CCP header version (CCPHDRVER) .....                                                         | 22 |
| 7.2.3 CCP address version (CCPADDRVER).....                                                        | 23 |
| 7.2.4 Destination address (DESTADDR).....                                                          | 23 |
| 7.2.5 Source address (SRCADDR) .....                                                               | 23 |
| 7.2.6 Type (TYPE) field .....                                                                      | 23 |
| 7.2.7 Reserved (RSV) field .....                                                                   | 25 |
| 7.2.8 CCP payload length (CCPPLEN) field.....                                                      | 25 |
| 7.2.9 CCP payload field .....                                                                      | 25 |
| 8 Home network management protocol (HNMP) .....                                                    | 25 |
| 8.1 General.....                                                                                   | 25 |
| 8.2 HNMP packet format .....                                                                       | 26 |
| 8.2.1 Transaction ID (TID) .....                                                                   | 26 |
| 8.2.2 HNMP command (HNMPCMD).....                                                                  | 26 |
| 8.2.3 Reserved (RSV) field .....                                                                   | 26 |

|        |                                                                          |    |
|--------|--------------------------------------------------------------------------|----|
| 8.2.4  | HNMP payload length (HNMPPLEN) field .....                               | 26 |
| 8.2.5  | HNMP payload .....                                                       | 26 |
| 8.3    | Home server registration .....                                           | 27 |
| 8.4    | Device registration .....                                                | 27 |
| 8.4.1  | Device registration request (DEV_REG_REQ) packet .....                   | 27 |
| 8.4.2  | Device registration response (DEV_REG_RES) packet .....                  | 28 |
| 8.5    | Device management .....                                                  | 29 |
| 8.5.1  | Add device (ADD_DEV) packet .....                                        | 31 |
| 8.5.2  | Delete device (DEL_DEV) packet .....                                     | 31 |
| 8.5.3  | Initialize device (INI_DEV) packet .....                                 | 32 |
| 8.5.4  | Alive-check request (ALV_CHK_REQ) packet .....                           | 32 |
| 8.5.5  | Alive-check response (ALV_CHK_RES) packet .....                          | 32 |
| 8.6    | Address and name information of devices .....                            | 32 |
| 8.6.1  | Device address and name information request (DEV_INFO_REQ) packet .....  | 33 |
| 8.6.2  | Device address and name information response (DEV_INFO_RES) packet ..... | 33 |
| 8.7    | Other management functions .....                                         | 34 |
| 9      | Universal home control protocol (UHCP) .....                             | 34 |
| 9.1    | UHCP packet format .....                                                 | 34 |
| 9.1.1  | Transaction ID (TID) .....                                               | 35 |
| 9.1.2  | Message type (MT) and action type (AT) .....                             | 35 |
| 9.1.3  | Reserved (RSV) field .....                                               | 36 |
| 9.1.4  | UHCP payload length (UHCPPLEN) field .....                               | 36 |
| 9.1.5  | UHCP payload .....                                                       | 36 |
| 9.2    | Execution messages (EXE) .....                                           | 36 |
| 9.2.1  | Execution of registration (EXE_REG) .....                                | 36 |
| 9.2.2  | Execution of control (EXE_CTRL) .....                                    | 37 |
| 9.2.3  | Response OK (EXE_RESOK) .....                                            | 38 |
| 9.2.4  | Response NOK (EXE_RESNOK) .....                                          | 38 |
| 9.3    | Query messages (QUE) .....                                               | 38 |
| 9.3.1  | Query of registration status (QUE_REGSTAT) .....                         | 38 |
| 9.3.2  | Query of control status (QUE_CTRLSTAT) .....                             | 39 |
| 9.3.3  | Query of all status (QUE_ALLSTAT) .....                                  | 39 |
| 9.3.4  | Response OK (QUE_RESOK) .....                                            | 40 |
| 9.3.5  | Response NOK (QUE_RESNOK) .....                                          | 40 |
| 9.4    | Notification messages (NTFY) .....                                       | 40 |
| 9.5    | UHCP payload syntax .....                                                | 40 |
| 9.5.1  | Basic syntax for UHCP payload .....                                      | 40 |
| 9.5.2  | Syntax for UHCP registration .....                                       | 41 |
| 9.5.3  | Syntax for device control .....                                          | 42 |
| 9.5.4  | Syntax for query of controlling and monitoring status .....              | 43 |
| 9.5.5  | Syntax for notification .....                                            | 44 |
| 10     | Home data service protocol (HDSP) .....                                  | 45 |
| 10.1   | Functional requirements of HDSP .....                                    | 45 |
| 10.1.1 | Interoperability with CCP .....                                          | 45 |
| 10.1.2 | File and directory services .....                                        | 45 |
| 10.1.3 | Messaging service .....                                                  | 46 |
| 10.2   | HDSP packet format .....                                                 | 46 |

|          |                                                                                                           |    |
|----------|-----------------------------------------------------------------------------------------------------------|----|
| 10.2.1   | Transaction ID (TID)                                                                                      | 46 |
| 10.2.2   | HDSP command                                                                                              | 46 |
| 10.2.3   | HDSP payload length (HDSPPLEN) field                                                                      | 47 |
| 10.2.4   | HDSP payload                                                                                              | 47 |
| 10.3     | Messages for directory services                                                                           | 47 |
| 10.3.1   | Query request message (DIR_QUE_REQ)                                                                       | 48 |
| 10.3.2   | Query response message (DIR_QUE_RES)                                                                      | 48 |
| 10.3.3   | Deletion request message (DIR_DEL_REQ)                                                                    | 49 |
| 10.3.4   | Deletion response message (DIR_DEL_RES)                                                                   | 49 |
| 10.3.5   | Renaming request message (DIR_REN_REQ)                                                                    | 49 |
| 10.3.6   | Renaming response message (DIR_REN_RES)                                                                   | 49 |
| 10.3.7   | Making request message (DIR_MAKE_REQ)                                                                     | 49 |
| 10.3.8   | Making response message (DIR_MAKE_RES)                                                                    | 50 |
| 10.4     | Messages for file services                                                                                | 50 |
| 10.4.1   | Query request message (FILE_QUE_REQ)                                                                      | 53 |
| 10.4.2   | Query response message (FILE_QUE_RES)                                                                     | 53 |
| 10.4.3   | Deletion request message (FILE_DEL_REQ)                                                                   | 53 |
| 10.4.4   | Deletion response message (FILE_DEL_RES)                                                                  | 53 |
| 10.4.5   | Renaming request message (FILE_REN_REQ)                                                                   | 53 |
| 10.4.6   | Renaming response message (FILE_REN_RES)                                                                  | 54 |
| 10.4.7   | Negotiation request message (FILE_NEGO_REQ)                                                               | 54 |
| 10.4.8   | Negotiation response message (FILE_NEGO_RES)                                                              | 54 |
| 10.4.9   | Getting request message (FILE_GET_REQ)                                                                    | 54 |
| 10.4.10  | Getting response message (FILE_GET_RES)                                                                   | 54 |
| 10.4.11  | Putting request message (FILE_PUT_REQ)                                                                    | 55 |
| 10.4.12  | Putting response message (FILE_PUT_RES)                                                                   | 55 |
| 10.5     | Messages for messaging service                                                                            | 55 |
| 10.5.1   | Sending request message (MSG_PUT_REQ)                                                                     | 56 |
| 10.5.2   | Sending response message (MSG_PUT_RES)                                                                    | 56 |
| 10.6     | Error codes                                                                                               | 56 |
| 11       | Home multimedia service protocol (HMSP)                                                                   | 57 |
| 11.1     | Functional requirements of HMSP                                                                           | 58 |
| 11.1.1   | Interoperability with CCP                                                                                 | 58 |
| 11.1.2   | Management of multimedia resource                                                                         | 58 |
| 11.1.3   | Stream and play of multimedia resource                                                                    | 58 |
| Annex A  | (informative) FSM of FS-CCPDEV supporting HNMP                                                            | 60 |
| Annex B  | (informative) FSM of FS-CCPDEV for supporting UHCP                                                        | 63 |
| Figure 1 | – Communication layer structures of network technologies                                                  | 10 |
| Figure 2 | – Server interface                                                                                        | 11 |
| Figure 3 | – Cluster and domain network                                                                              | 12 |
| Figure 4 | – Classification of CCP devices                                                                           | 16 |
| Figure 5 | – Definitions of application program, CCP API, lower protocol layers interface, and lower protocol layers | 18 |
| Figure 6 | – Location of CCP layer                                                                                   | 18 |
| Figure 7 | – Example of data transmission over heterogeneous networks using CCP layer                                | 20 |

|                                                                                                       |    |
|-------------------------------------------------------------------------------------------------------|----|
| Figure 8 – CCP address format of CCP address by version 0 .....                                       | 21 |
| Figure 9 – CCP packet format of CCP header by version 0 .....                                         | 22 |
| Figure 10 – Type fields .....                                                                         | 23 |
| Figure 11 – HNMP packet format .....                                                                  | 26 |
| Figure 12 – DEV_REG_REQ and DEV_REG_RES packets .....                                                 | 28 |
| Figure 13 – Example of HNMP command sequence for device registration .....                            | 29 |
| Figure 14 – ADD_DEV, DEL_DEV and INI_DEV packets .....                                                | 30 |
| Figure 15 – ALV_CHK_REQ and ALV_CHK_RES packets .....                                                 | 30 |
| Figure 16 – Example of HNMP command sequence for device management .....                              | 31 |
| Figure 17 – DEV_INFO_REQ and DEV_INFO_RES packets .....                                               | 33 |
| Figure 18 – Example of HNMP command sequence for retrieving device address and name information ..... | 33 |
| Figure 19 – UHCP packet format .....                                                                  | 35 |
| Figure 20 – Message type and action type fields of UHCP packet .....                                  | 36 |
| Figure 21 – Example of registration process .....                                                     | 37 |
| Figure 22 – Example of EXE_CTRL message .....                                                         | 38 |
| Figure 23 – Example of QUE_REGSTAT message .....                                                      | 39 |
| Figure 24 – Example of QUE_CTRLSTAT message .....                                                     | 39 |
| Figure 25 – Example of QUE_ALLSTAT message .....                                                      | 40 |
| Figure 26 – HDSP packet format .....                                                                  | 46 |
| Figure 27 – Example of usage of directory service messages .....                                      | 50 |
| Figure 28 – Example of usage of file service messages .....                                           | 52 |
| Figure 29 – Example of usage of Messaging service messages .....                                      | 56 |
| Figure A.1 – FSM of FS-CCPDEV for supporting HNMP .....                                               | 60 |
| Figure B.1 – FSM of FS-CCPDEV for supporting UHCP .....                                               | 63 |
| Table 1 – Cast type field .....                                                                       | 24 |
| Table 2 – Traffic type field .....                                                                    | 24 |
| Table 3 – Payload type field .....                                                                    | 25 |
| Table 4 – HDSP commands .....                                                                         | 47 |
| Table 5 – Messages for directory services .....                                                       | 48 |
| Table 6 – Messages for file services .....                                                            | 51 |
| Table 7 – Messages for messaging services .....                                                       | 55 |
| Table 8 – Error codes for HDSP .....                                                                  | 57 |

## INTERNATIONAL ELECTROTECHNICAL COMMISSION

# MULTIMEDIA SYSTEMS – COMMON COMMUNICATION PROTOCOL FOR INTER-CONNECTIVITY ON HETEROGENEOUS NETWORKS

## 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 provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
- 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.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62295 has been prepared by technical area 8: Multimedia home server systems, of IEC technical committee 100: Audio, video and multimedia systems and equipment.

This bilingual version (2012-08) corresponds to the monolingual English version, published in 2007-11.

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

| CDV          | Report on voting |
|--------------|------------------|
| 100/1200/CDV | 100/1283/RVC     |

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

The French version of this standard has not been voted upon.

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

The committee has decided that the contents of this publication will remain unchanged until the maintenance result 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.

**IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.**

IECNORM.COM : Click to view the full PDF of IEC 62295:2007

## INTRODUCTION

Numerous wired and wireless home network technologies of various types have been developed and are in use today. However, since these technologies have been developed for specific functions such as control, A/V and data services, interoperability is not guaranteed among products employing these technologies. Hence, users who wish to implement the home network environment either purchase devices that are based on a single technology for interoperability or install independent, non-compatible networks in their home.

To solve these problems, home network businesses and service providers have taken into account and developed a number of specific technologies in order to allow interoperability among home network technologies. However, most of these technologies are local and offer interoperability between a limited range of devices and give rise to new problems caused by complexity and diversity in technologies of different companies.

In order to incorporate such complex and diverse technologies, there is a need to develop a new convergence technology that can integrate not only current technologies but also those expected to surface in the future.

The needs for the new convergence technology are the following:

- provide interoperability and interconnectivity among heterogeneous networks through a specific convergence layer;
- provide expandability for applications in not only current network technologies, but also new technologies to be developed in the future;
- are applicable in small devices with low processing capabilities by providing protocols such as simple signaling in the convergence layer;
- available at a low cost and simple to implement on a device;
- able to provide diverse home network services (or applications).

## **MULTIMEDIA SYSTEMS – COMMON COMMUNICATION PROTOCOL FOR INTER-CONNECTIVITY ON HETEROGENEOUS NETWORKS**

### **1 Scope and object**

This International Standard specifies the common communication protocol (CCP) layer that is capable of providing interoperability and interconnectivity between heterogeneous network technologies, as well as the basic data transmission scheme between devices linked to heterogeneous networks through the CCP layer. The standard also specifies the packet structure in the CCP layer and the common addressing scheme that can be understood among heterogeneous devices. Furthermore, there are specifications regarding protocols capable of providing diverse home network applications through the CCP layer such as the home network management protocol (HNMP), universal home control protocol (UHCP), home multimedia service protocol (HMSP) and home data service protocol (HDSP).

NOTE HNMP is the overall home network management protocol that detects or registers devices. UHCP controls and monitors devices from remote locations. HMSP is the A/V protocol for home entertainment services. HDSP deals with data and messaging services.

This standard is to be applied to systems with network capabilities and those that constitute home networks such as electronic appliances, A/V components, control devices, network terminals and home servers. Moreover, this standard is applicable to a home network consisting of a single home server.

This International Standard gives

- a definition of the CCP layer,
- a data transmission scheme in the CCP layer,
- a CCP packet structure,
- a CCP addressing scheme,
- a home network management protocol (HNMP),
- a universal home control protocol (UHCP),
- a home data service protocol (HDSP),
- requirements of home multimedia service protocol (HMSP).

A home network provides interoperability and interconnectivity regardless of the appliance manufacturer or the network type so that the user is able to receive desired services at any point in time. However, current home network technologies have independent communication protocol layer structures, as shown in Figure 1, with different addressing schemes, data transmission schemes, data processing methods and data formats.

In order to solve problems associated with interoperability and interconnectivity among heterogeneous network technologies, this standard aims to define the CCP layer as a type of a convergence layer.

Further objectives of this standard include specifying the data transmission method, packet structure and common addressing scheme as well as the signaling protocol for providing home network management, control, A/V and data services.

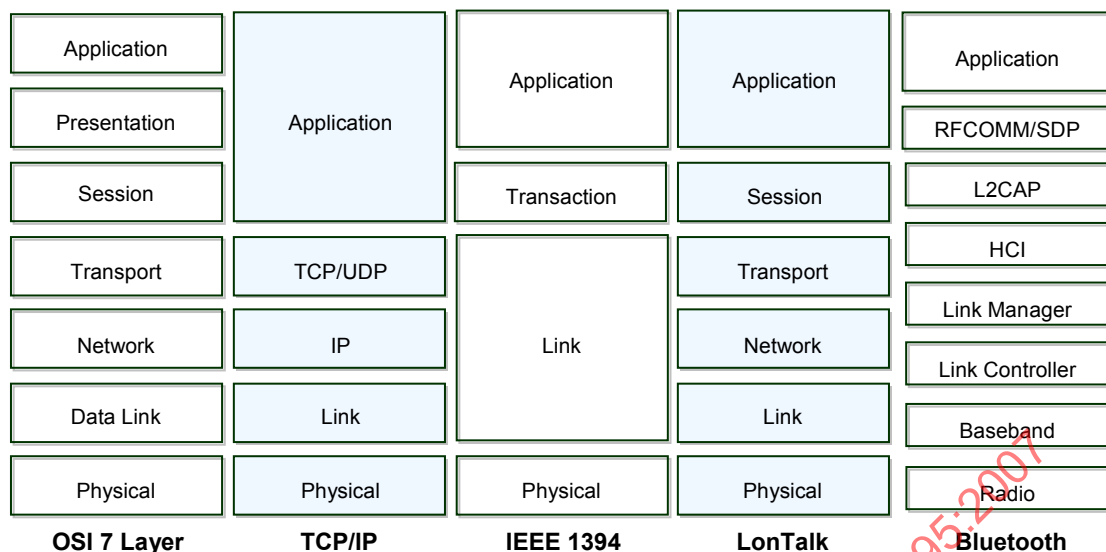


Figure 1 – Communication layer structures of network technologies

IEC 2072/07

## 2 Normative references

None.

## 3 Terms, definitions, abbreviations and conventions

For the purposes of this document, the following definitions apply.

### 3.1 Terms and definitions

#### 3.1.1

##### CCP device

device which has networking capabilities using commercial network technologies to link to a home network; it also supports at least home network management protocol (HMNP) among four protocols provided in the CCP layer and CCP

#### 3.1.2

##### CCP addressing

common addressing scheme used in the CCP layer which consists of four fields: domain address, cluster address and device ID

NOTE One of the characteristics of CCP addressing is that it has a structure that is understood by application programs embedded in devices linked to heterogeneous networks as well as the device users.

#### 3.1.3

##### home server interface

##### HSI

interface module in a home server responsible for connection with a particular network which can process the corresponding physical interface and the communication protocol according to the type of network it is connected to

NOTE A home server interface is shown in Figure 2.

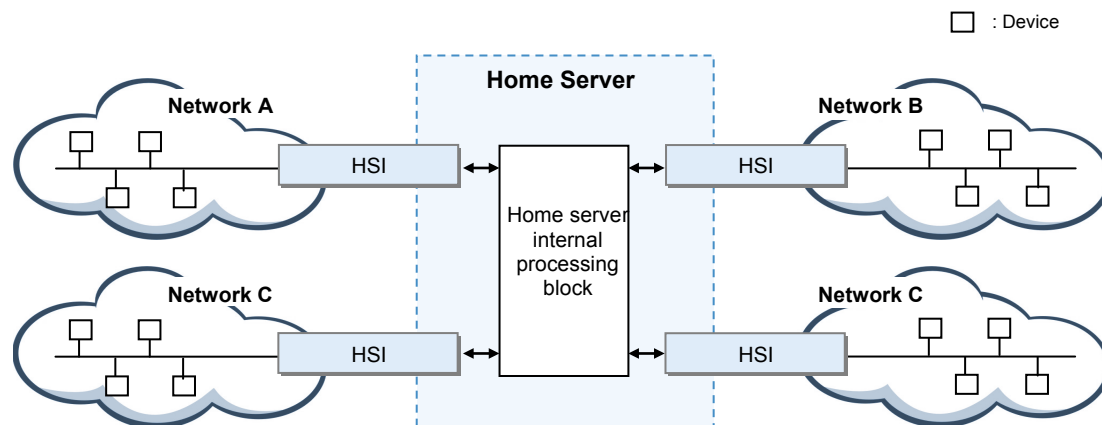


Figure 2 – Server interface

IEC 2073/07

**3.1.4****cluster network**

collection of devices using an identical physical interface and communication protocol

NOTE 1 As shown in Figure 3, a cluster network is a collection of devices using an identical physical interface and communication protocol. Moreover, a HSI that links a home server to the devices within the cluster network is also part of the cluster network. A cluster address is one of the CCP addressing fields defined in the CCP layer, and it is used to distinguish multiple cluster networks linked to a single home server.

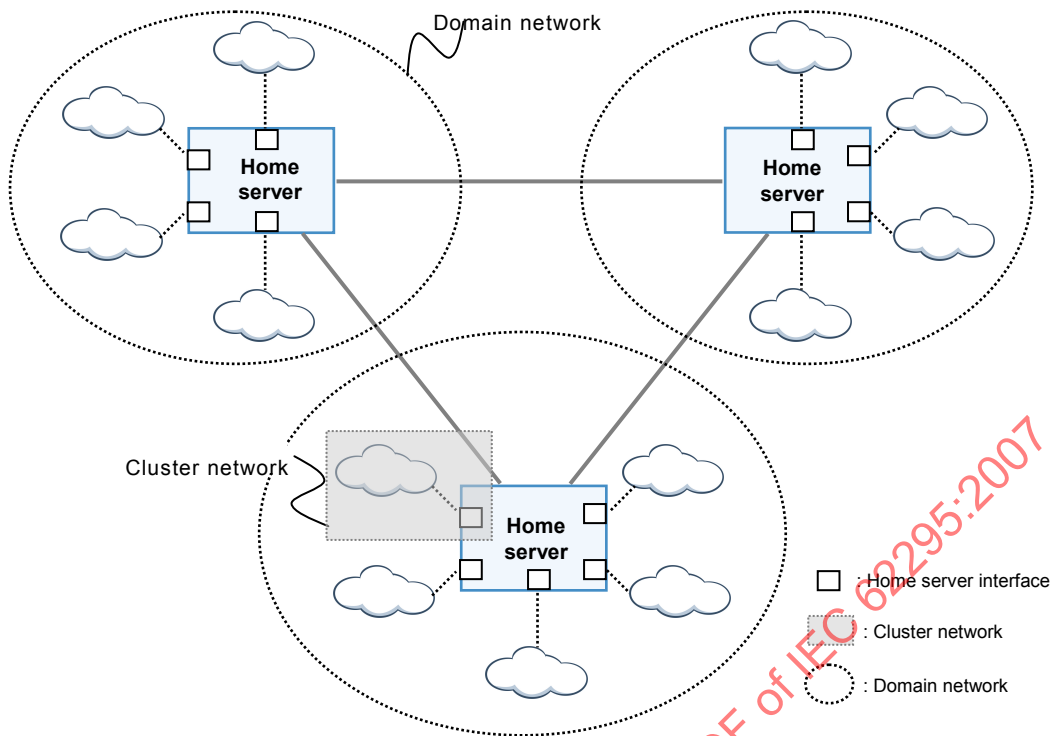
NOTE 2 A HSI that links a home server to the devices within the cluster network is also part of the cluster network.

NOTE 3 A cluster network is shown in Figure 3.

**3.1.5****cluster address**

one of the CCP addressing fields defined in the CCP layer which is used to distinguish multiple cluster networks linked to a single home server

NOTE A cluster address is shown in Figure 3.



IEC 2074/07

**Figure 3 – Cluster and domain network**

### 3.1.6

#### **domain network**

collection of devices connected to a single home server

NOTE 1 Devices within the multiple cluster networks linked to the home server and the HSIs managing the cluster network interface constitute the domain network.

NOTE 2 A domain network is shown in Figure 3.

### 3.1.7

#### **domain address**

one of the CCP addressing fields defined in the CCP layer which is used to distinguish multiple domain networks in the home.

NOTE A domain address is shown in Figure 3.

### 3.1.8

#### **single-HS network**

a home network comprising a single home server where there is only one domain network under the single-HS network environment

### 3.1.9

#### **multi-HS network**

a home network with two or more home servers where there are multiple domain networks under the multi-HS network environment

NOTE A multi-HS network is shown in Figure 3.

**3.1.10****CCP application program**

a CCP application program is positioned at the top of the CCP layer. A CCP application program sends and receives data to/from CCP layer through CCP application programming interface

**3.1.11****application program data unit****APDU**

the unit of data exchanged among multiple application programs on an equivalent level

**3.1.12****CCP application programming interface(API)**

the interface between the application program and the CCP layer

**3.1.13****lower protocol layers**

lower protocol layers refer to all communication protocol layers below the CCP layer, with the exception of the physical layer

NOTE A diagram of lower protocol layers is shown in Figure 5.

**3.1.14****lower protocol layers interface**

the interface between the APDU and lower protocol layers which are referred to as the lower protocol layers interface

NOTE A diagram of lower protocol layers interface is shown in Figure 5.

**3.2 Abbreviations**

|            |                                     |
|------------|-------------------------------------|
| ACDS       | Active Control & Data Service       |
| ACMS       | Active Control & Multimedia Service |
| ACS        | Active Control Service              |
| ALLSTAT    | All Status                          |
| APDU       | Application Program Data Unit       |
| API        | Application Program Interface       |
| AT         | Action Type                         |
| ATTR       | Attribute                           |
| CCP        | Common Communication Protocol       |
| CCPADDRVER | CCP Address Version                 |
| CCPDEV     | CCP Device                          |
| CCPH       | CCP Header                          |
| CCPHDRVER  | CCP Header Version                  |
| CCPID      | CCP Identification                  |
| CCPLEN     | CCP Payload Length                  |
| CMD        | Command                             |
| CT         | Cast Type                           |
| CTRLSTAT   | Control Status                      |
| DATALEN    | Data Length                         |
| DEV        | Device                              |
| DMS        | Data & Multimedia Service           |
| DS         | Data Service                        |

|          |                                  |
|----------|----------------------------------|
| DSTADDR  | Destination Address              |
| ERRCODE  | Error Code                       |
| FMTS     | Final Maximum Transfer Size      |
| FS       | Full Service                     |
| FSM      | Finite State Machine             |
| HDSP     | Home Data Service Protocol       |
| HDSPPLEN | HDSP Payload Length              |
| HMSP     | Home Multimedia Service Protocol |
| HNMP     | Home Network Management Protocol |
| HNMPCMD  | HNMP Command                     |
| HNMPPLEN | HNMP Payload Length              |
| HS       | Home Server                      |
| HSI      | Home Server Interface            |
| MCS      | Monitoring Control Service       |
| MON      | Monitoring                       |
| MS       | Multimedia Service               |
| MT       | Message Type                     |
| MTS      | Maximum Transfer Size            |
| MYMTS    | My Maximum Transfer Size         |
| PATHLEN  | Path Length                      |
| PCS      | Passive Control Service          |
| PnP      | Plug-and-Play                    |
| PT       | Payload Type                     |
| QoS      | Quality of Service               |
| REG      | Registration                     |
| REGSTAT  | Registration Status              |
| RES      | Response                         |
| SEC      | Second                           |
| SEGNO    | Segmentation Number              |
| SEQNO    | Sequence Number                  |
| SRCADDR  | Source Address                   |
| STAT     | Status                           |
| TID      | Transaction ID                   |
| TT       | Traffic Type                     |
| UHCP     | Universal Home Control Protocol  |
| UHCPPLEN | UHCP Payload Length              |

### 3.3 Conventions

UHCP payload content is expressed as italic character as follows:

E.g. <ITEM>*argument*</ITEM>

## 4 Requirements

This clause describes the requirements in order to implement the HSI and CCP devices.

## 4.1 Home server interface requirements

The HSI shall satisfy the following requirements.

### 4.1.1 Basic requirements for data delivery

- In order to support the device registration process, the HSI shall be able to receive all packets being broadcast to the cluster network from the physical layer and relay them to the CCP layer.
- The HSI shall maintain and manage an address table that contains the CCP addresses of CCP devices in its cluster network as well as the physical addresses used in the cluster network.
- Upon completion of the device registration process, when delivering packets to a particular CCP device within its cluster network, the HSI shall be able to transmit packets from the physical layer using the CCP device's physical address that corresponds to the CCP destination address.
- An HSI shall support the CCP layer as well as the requirements specified in 4.1. HSI can be implemented with the card module-type for the physical interface, communication protocol and the CCP layer.

### 4.1.2 Functional requirements for HNMP

- After receiving a Device Registration Request (DEV\_REG\_REQ) packet, the HSI shall be able to send a Device Registration Response (DEV\_REG\_RES) packet to the CCP device from which the DEV\_REG\_REQ packet originated.
- The HSI shall be able to store CCP and physical addresses of the CCP device contained in the DEV\_REG\_REQ packet in the address table.
- The HSI shall be able to allocate a unique Device ID to the CCP device from which the DEV\_REG\_REQ packet originated. The HSI shall also specify data such as the allocated Device ID, domain address and the HSI's own physical address in DEV\_REG\_RES packet's HNMP payload prior to transmission.
- The HSI shall be able to notify the fact that a new CCP device has been registered in the home network to all CCP devices within its cluster network as well as all other HSIs in its domain network by transmitting an Add Device (ADD\_DEV) packet.
- In order to support CCP device's Plug-and-Play (PnP) function, the HSI shall send on a regular basis an Alive-Check Request (ALV\_CHK\_REQ) packet to the CCP devices that have completed the device registration process and assess the response packets from them. The HSI shall be able to determine that the non-responsive CCP devices have lost connection with the home network.
- The HSI shall be able to notify the fact that a particular CCP device has been disconnected from the home network to all CCP devices within its cluster network as well as all other HSIs in its domain network by transmitting a Delete Device (DEL\_DEV) packet.

### 4.1.3 Home server interface requirements for unicast, multicast and broadcast

- The HSI shall be able to process all packets received at its physical layer from the physical layer to lower protocol layers before sending them to the CCP layer.
- The HSI shall be able to examine the cluster address in the CCP destination address field of a packet arriving at the CCP layer through lower protocol layers and send the packet to the HSI responsible for the interface of the cluster network that corresponds to the cluster address.
- When a CCP packet arrives from another HSI
  - if the cast type of the received CCP packet is unicast and the CCP destination address is specified, then the HSI receiving the packet shall be able to relay the packet to the particular CCP device within its cluster network,
  - if the cast type of the received CCP packet is multicast, then the HSI receiving the packet shall be able to relay the packet to only the restricted number of the CCP devices within its cluster network in accordance with the content of the multicast,

- if the cast type of the received CCP packet is broadcast, then the HSI receiving the packet shall be able to relay the packet to all CCP devices within its cluster network.

## 4.2 CCP device requirements

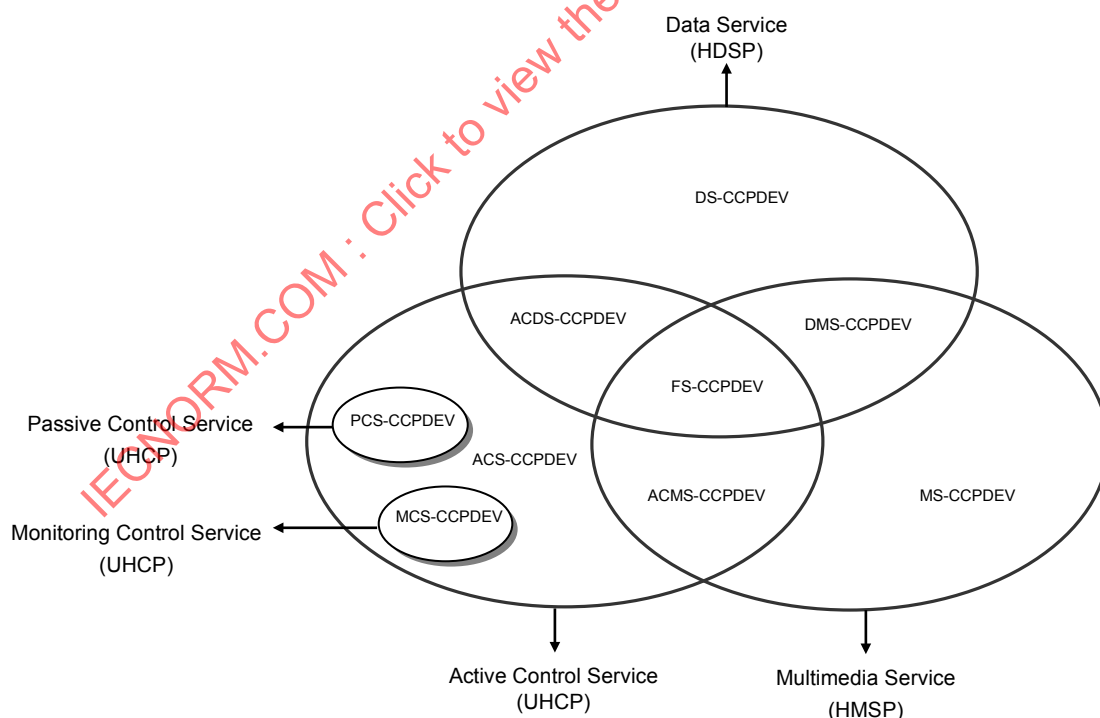
This subclause specifies the requirements for the device registration process needed for CCP devices to perform the plug-and-play function, as well as the requirements for classifying CCP devices according to the CCP functions provided by CCP devices.

### 4.2.1 Requirements for device registration

- If a CCP device is not aware of the physical address of the HSI of its own cluster network, the HSI's physical address (or network address) shall be obtained through the device registration process.
- For the purpose of device registration, a CCP device shall be able to initially broadcast a device registration request (DEV\_REG\_REQ) packet to its own cluster network.
- Upon completion of the device registration process, when a CCP device transmits data to a heterogeneous network, it shall send all packets from the physical layer to the HSI of its own cluster network, regardless of the cast type and the CCP destination address.

### 4.2.2 Requirements for classification of CCP devices

CCP devices refer to devices that support HNMP among the functions provided by the CCP layer and the CCP. CCP devices can be classified according to their processing capabilities and the network technologies being used. This standard shall classify CCP devices into nine categories according to the functions provided by the CCP devices, as shown in Figure 4.



IEC 2075/07

Figure 4 – Classification of CCP devices

- Active Control Service CCP Device (ACS-CCPDEV) is a CCP device that only supports the UHCP function.

- Passive Control Service CCP Device (PCS-CCPDEV) is a CCP device whose UHCP function is carried out by accepting commands from other devices.
- Monitoring Control Service CCP Device (MCS-CCPDEV) is a CCP device that only provides the monitoring capabilities among UHCP functions.
- Data Service CCP Device (DS-CCPDEV) is a CCP device that only provides HDSP.
- Multimedia Service CCP Device (MS-CCPDEV) is a CCP device that only provides the HMSP function.
- Active Control & Data Service CCP Device (ACDS-CCPDEV) is a CCP device that provides UHCP and HDSP functions.
- Active Control & Multimedia Service CCP Device (ACMS-CCPDEV) is a CCP device that provides UHCP and HMSP.
- Data & Multimedia Service CCP Device (DMS-CCPDEV) is a CCP device that provides HDSP and HMSP.
- Full Service CCP Device (FS-CCPDEV) is a CCP device that provides UHCP, HDSP and HMSP functions, and it is also equipped with a display panel for display capabilities.

## 5 Common communication protocol (CCP) layer

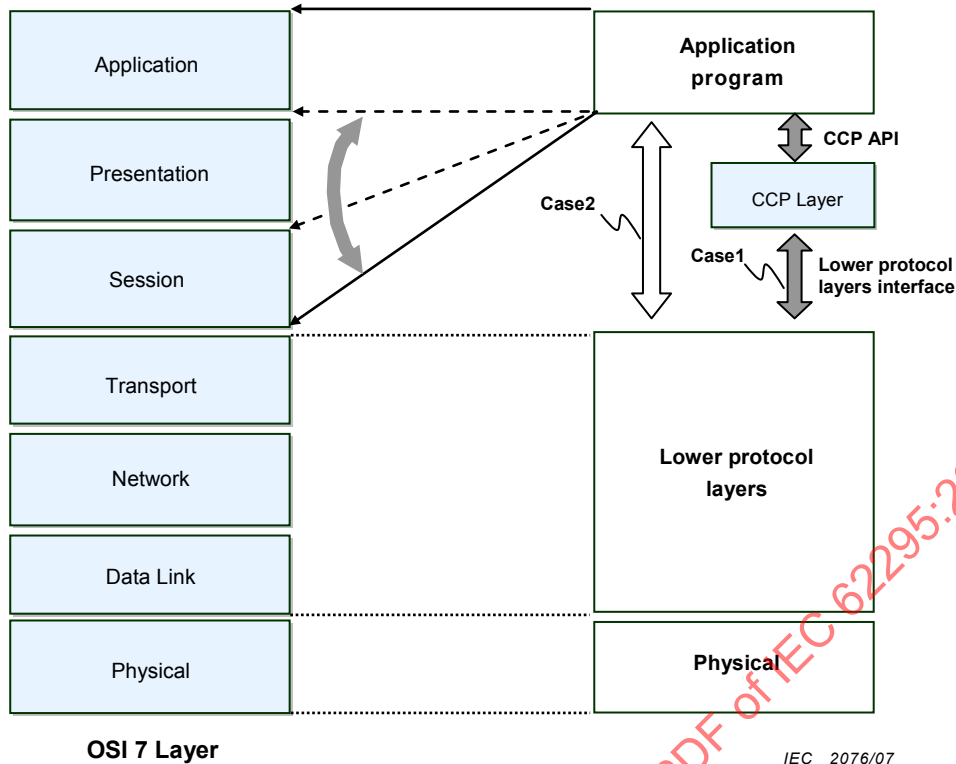
This clause specifies the CCP layer, which allows bilateral communication among heterogeneous networks of various types of application programs.

### 5.1 CCP layer

The CCP layer provides a common communication channel for devices (or applications executed from the devices) linked to heterogeneous networks. This CCP layer shall be located below all application programs provided by the service providers (or device manufacturers). From OSI 7 layer's perspective, although most network protocols and application programs are similar to the OSI 7 layer structure, they are not clearly divided into seven layers. The location of the CCP layer is variable according to the layer covered by the application program (or the structure of lower protocol layers).

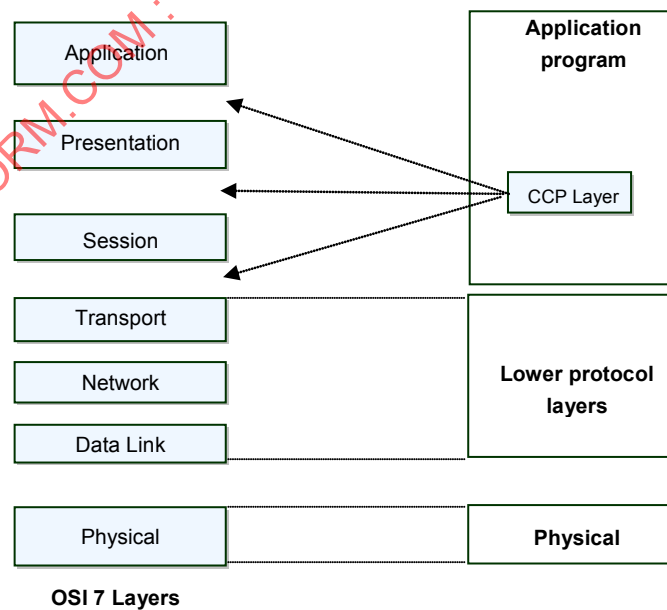
Case 1 in Figure 5 is an example where an application program of a home network device sends data to lower protocol layers through the CCP layer, whereas Case 2 displays an example of an application program of a home network device directly sending data to lower protocol layers without going through the CCP layer.

The application program in Case 1 is transmitting a desired Application Program Data Unit (APDU) to a device connected to a cluster network linked to a heterogeneous network via the home server. On the other hand, the Case 2 example can be used when the application program is to send a desired Application Program Data Unit (APDU) to a device located in the same cluster network using the same network protocol, bypassing the home server. Therefore, in Case 2, all services that have been offered within the same network prior to applying the CCP layer (such as FTP among TCP/IP networks and Telnet service) remain available.



**Figure 5 – Definitions of application program, CCP API, lower protocol layers interface, and lower protocol layers**

In case of an application program only covering up to the application layer as shown in Figure 6, the CCP layer is located below the application layer. For an application program covering from the application layer to the presentation layer, the CCP layer is located below the presentation layer. For an application program covering from the application layer to the session layer, the CCP layer is located below the session layer.



**Figure 6 – Location of CCP layer**

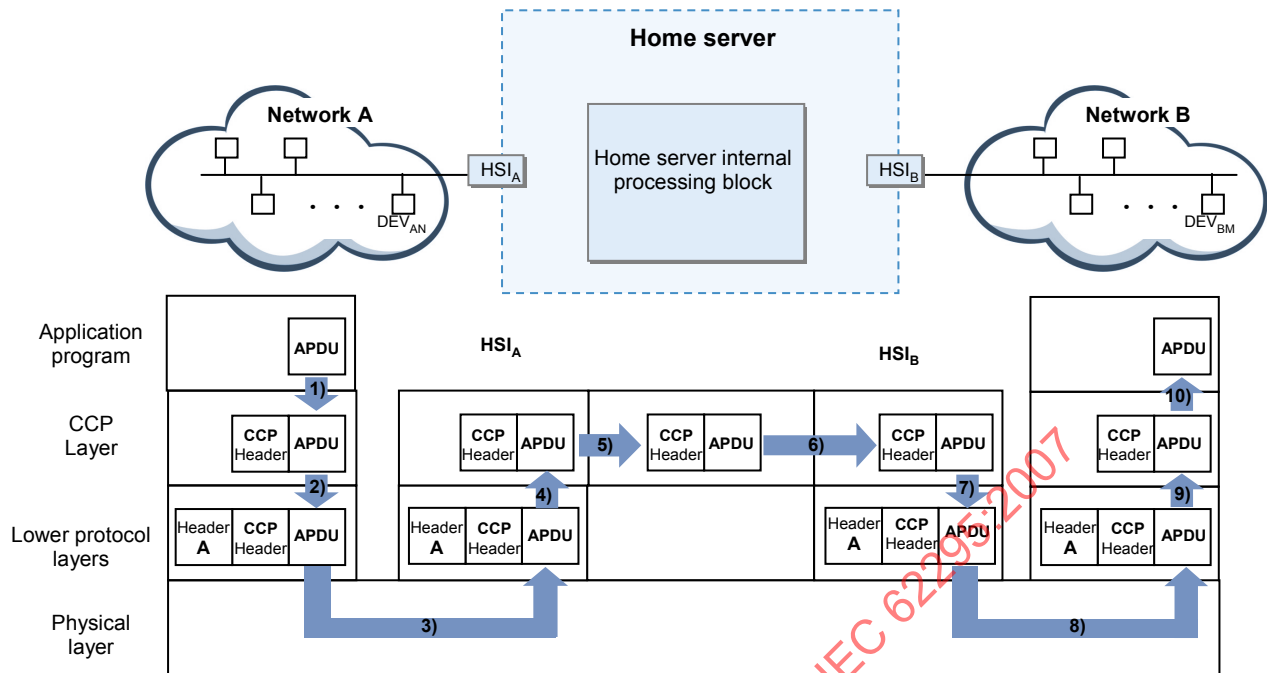
## 5.2 Data delivery over heterogeneous networks

This subclause describes the data transmission process using Figure 7. For data communication with a heterogeneous network, an APDU shall go through the CCP layer before being sent to lower protocol layers. For a home network device to transmit an APDU to a heterogeneous network, the application program sends an APDU to the CCP layer through the CCP API, and the CCP layer encapsulates the received APDU with a CCP header and relays it to lower protocol layers.

- DEV<sub>An</sub>: CCP Device in Network A with Device ID n
- DEV<sub>Bm</sub>: CCP Device in Network B with Device ID m
- HSI<sub>A</sub>, HSI<sub>B</sub>: HSI responsible for interface with Network A (or Network B) in Home server.

The data transmission process is as follows.

- 1) The application program of DEV<sub>An</sub> sends an APDU to the CCP layer through the CCP API.
- 2) The CCP layer of DEV<sub>An</sub> encapsulates the APDU from the application program with a CCP header and sends it to lower protocol layers through the lower protocol layers interface.
- 3) Lower protocol layers generate a header for each corresponding layer, and the physical layer designates the physical address of HSI<sub>A</sub> as the destination address and transmits the packet.
- 4) The physical layer of HSI<sub>A</sub> receives the packet and relays it to lower protocol layers. The Lower Protocol Layers process a header for each corresponding layer and send it to the CCP layer.
- 5) The CCP layer of HSI<sub>A</sub> relays the arrived CCP packet to the home server internal processing block of the home server.
- 6) The home server internal processing block analyzes the received packet's CCP header information to process the packet and sends it to the HSI(HSI<sub>B</sub>) corresponding to the cluster address of the CCP destination address.
- 7) The CCP layer of HSI<sub>B</sub> transmits the received packet to lower protocol layers. When doing so, HSI<sub>B</sub> shall search for and retrieve the physical address of DEV<sub>Bm</sub> corresponding to the CCP destination address from the address table and notify it to the physical layer.
- 8) The physical layer of HSI<sub>B</sub> designates the physical address of DEV<sub>Bm</sub> as the destination address and transmits the packet.
- 9) The physical layer of DEV<sub>Bm</sub> receives the packet and sends it to lower protocol layers, where each header is processed corresponding to each layer and the packet is transmitted to the CCP layer.
- 10) The CCP layer of DEV<sub>Bm</sub> delivers the arrived packet to the application program through the CCP API.



IEC 2078/07

**Figure 7 – Example of data transmission over heterogeneous networks using CCP layer**

## 6 CCP addressing

### 6.1 General

In order to implement intercommunication and interoperability under the home network environment where diverse heterogeneous networks coexist, devices that constitute the home network shall be seen to one another as being linked to a single network regardless of the type of network each device is connected to. This idea can only be materialized if there is an addressing scheme that can be understood between devices of different networks. Furthermore, an addressing scheme that can be understood by humans and application programs will facilitate its use and make implementation of application programs easy.

This clause specifies a new CCP addressing scheme according to the following three functional requirements:

- A new common address structure adequate for the home network environment.
- An addressing mechanism that can be understood by the user as well as the application program.

### 6.2 An addressing structure to facilitate traffic switching for the home server. Version 0

This subclause describes the CCP address format of version 0. The version field related to the CCP addressing is described in Clause 8. The CCP address format can be changed or extended for future use as other protocols that have their own addressing scheme. Anyway, all CCP devices of CCP address version 0 shall meet the requirements described in 7.2.

In order to satisfy the functional requirements, the CCP address format of version 0 is defined with four bytes, as shown in Figure 8. The CCP address contains three fields: Domain Address, Cluster Address and Device ID.

|                            |                             |                       |
|----------------------------|-----------------------------|-----------------------|
| Domain address<br>(1 byte) | Cluster address<br>(1 byte) | Device ID<br>(2 byte) |
|----------------------------|-----------------------------|-----------------------|

**Figure 8 – CCP address format of CCP address by version 0**

### 6.2.1 Domain address

The domain address distinguishes the domain network. When a home network in an office (or home) consists of more than one home server, the domain address is used to differentiate between them. Since one domain network constitutes a single home server, the domain address can be referred to as the home server ID.

The domain address consists of one byte. Therefore, there can be 256 domain addresses (from 0 to 255). Under the single-HS network environment, the domain address of 1 shall be used. Since a device does not know its own domain address before finishing the device registration process described in 9.3, the device shall initially use domain address '0'.

Under the multi-HS network environment where multiple home servers exist, there needs to be a process for allocating a domain address to each home server through the HNMP home server registration procedure. However, this document version does not consider the standards for home server registration under the multi-HS network environment.

### 6.2.2 Cluster address

The cluster address distinguishes multiple cluster networks linked to a single domain network. Since a single cluster network is linked to a single HSI, the cluster address can be used to switch the packets received within the home server. In order to denote the cluster address as a unique value, the cluster address shall be defined in this standard as the port number of a HSI. Therefore, there can also be 256 cluster addresses (from 0 to 255). Since a device does not know its own cluster address, either before finishing device the registration process described in 9.3, the device shall initially use cluster address '0'.

### 6.2.3 Device ID field

A device ID field is used to distinguish CCP devices with the same domain and cluster address. Initially, a CCP device is not aware of its device ID. Therefore, each CCP device shall be allocated with its device ID from the HSI through the device registration process, which is one of HNMP functions.

When carrying out the device registration process, the HSI shall be able to assign different device IDs to CCP devices with an identical domain and cluster address.

The device ID is a two-byte field. Therefore, the range of device ID is from 0 to 65,535. Before finishing the device registration process, a device shall initially use device ID '0', and the HSI shall assign device IDs from 1 to 65,535 sequentially, whenever devices register.

## 7 CCP packet format and fields

### 7.1 General

This chapter describes the CCP packet structure used in the CCP layer as well as each field used in the CCP packet.

The CCP provides the Home Network Management Protocol (HNMP) for basic home network management services, the Universal Home Control Protocol (UHCP) for device control and

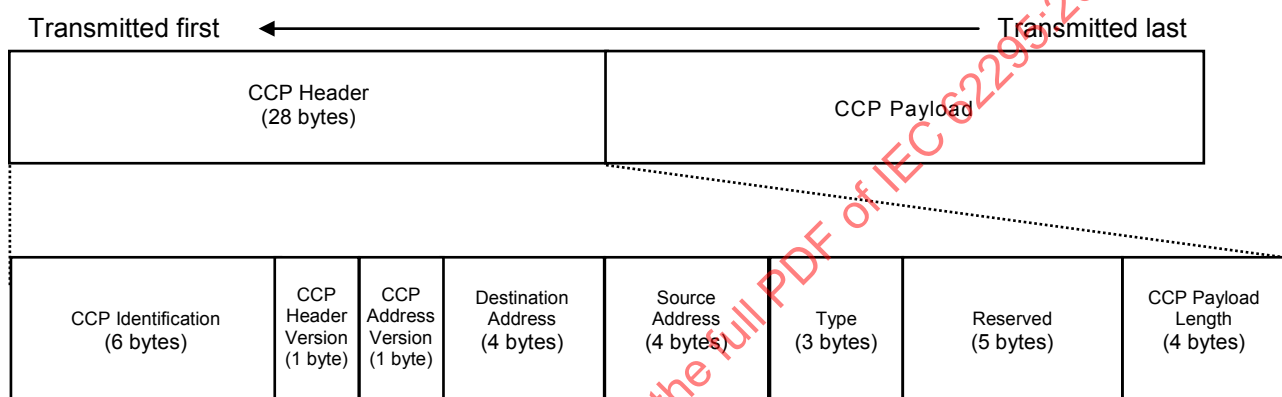
monitoring services, the Home Data Service Protocol (HDSP) for data and messaging services, and the Home Multimedia Service Protocol (HMSP) for multimedia services.

All of the protocols mentioned above (with the exception of UHCP) are signaling protocols based on messages defined in this regulation.

## 7.2 CCP packet format

Figure 9 shows the CCP packet format of version 0. A CCP packet largely consists of a CCP header and a CCP payload. The CCP header is 28 bytes long, containing CCP Identification, Destination Address, Source Address, Type Field, Reserved Field and CCP Payload Length Field.

The CCP payload comprises the APDUs received from the application program.



IEC 2079/07

**Figure 9 – CCP packet format of CCP header by version 0**

### 7.2.1 CCP identification (CCPID)

CCPID is used to determine whether a packet received at the CCP layer is a CCP packet or not.

If it is a CCP packet, its CCP header is processed and the CCP payload is sent to the application program. If it is not a CCP packet, the packet received from lower protocol layers is transmitted directly to the application program.

Henceforth, the CCP device can communicate with other CCP devices on heterogeneous networks as well as the ones of the same cluster network.

CCPID is six bytes long. The CCPID field is defined as text string of "IECccp", signifying a CCP header as an ASCII code value (0x494543636370). The CCP layer shall initially check the first four bytes of a packet received from lower protocol layers to assess whether it is a CCP packet.

### 7.2.2 CCP header version (CCPHDRVER)

Since each field in CCP header can be changed or extended for future use, CCPHDRVER is defined after the CCPID. The length of the CCPHDRVER field is one byte, and it shall be set to 0x00 in case of version 0. This version number will be increased by one if there is any modification or change in the CCP header format and size.

### 7.2.3 CCP address version (CCPADDRVER)

Since each field in CCP address can be changed or extended for future use, CCPADDRVER is defined after the CCPHDRVER. The length of CCPADDRVER field is one byte, and it shall be set to 0x00 in case of version 0. This version number will also be increased by one if there is any modification or change in the CCP address format and size.

### 7.2.4 Destination address (DESTADDR)

DADDR indicates the CCP address of the destination CCP device to which the CCP packet is to be delivered. In case that CCPADDRVER is 0, the destination address is four bytes long.

As specified in this clause, a destination address consists of three fields: Domain Address, Cluster Address and Device ID.

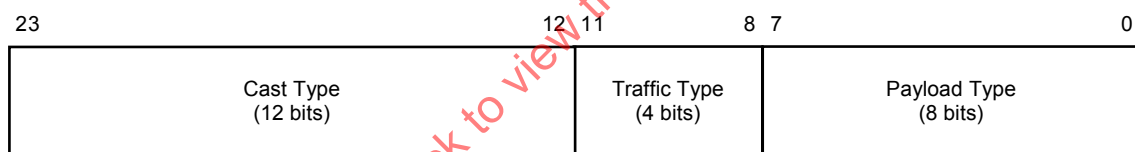
### 7.2.5 Source address (SRCADDR)

SADDR indicates the CCP address of the CCP device transmitting the packet. In case that CCPADDRVER is 0, the source address is four bytes long and also composed of three fields: Domain Address, Cluster Address and Device ID.

### 7.2.6 Type (TYPE) field

The type fields consist of three fields that indicate the packet delivery method, traffic type and payload type: Cast Type, Traffic Type and Payload Type. The Type fields are three bytes long.

Figure 10 displays the Type fields.



IEC 2080/07

**Figure 10 – Type fields**

#### 7.2.6.1 Cast type (CT)

The cast type field defines how the source CCP device delivers a packet via the home server. It is a 12-bit field.

As shown in Table 1, this standard provides seven methods: Unicast, Cluster-Multicast, Domain-Multicast, Device-Multicast, Device-Broadcast, HS-Broadcast and Broadcast.

**Table 1 – Cast type field**

| Cast type[11:0]                                                                                    | Notation          | Description                                                                                                                                                                                     |
|----------------------------------------------------------------------------------------------------|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x0XX <sup>a</sup>                                                                                 | Unicast           | Used for 1:1 communication between CCP devices.                                                                                                                                                 |
| 0x1WW <sup>b</sup>                                                                                 | Domain-Multicast  | Used for transmitting packets to HSIs, home servers and all CCP devices within a domain network with the domain address WW.                                                                     |
| 0x2YY <sup>c</sup>                                                                                 | Cluster-Multicast | Used for transmitting packets to all CCP devices including HSIs within the cluster network with the cluster address YY.                                                                         |
| 0xF0F                                                                                              | Device-Broadcast  | Used when there are multiple domain networks, such as the multi-HS network environment, for transmitting packets only to CCP devices within the domain network.                                 |
| 0xFF0                                                                                              | HS-Broadcast      | Used when there are multiple domain networks, such as the multi-HS network environment, for transmitting packets to all home servers and HSIs within the domain network but not to CCP devices. |
| 0xFFF                                                                                              | Broadcast         | Used when there are multiple domain networks, such as the multi-HS network environment, for transmitting packets to all HSIs, home servers and CCP devices within the domain network.           |
| Others                                                                                             | reserved          | Reserved for future usage.                                                                                                                                                                      |
| <sup>a</sup> XX: Don't care<br><sup>b</sup> WW: Domain address<br><sup>c</sup> YY: Cluster address |                   |                                                                                                                                                                                                 |

#### 7.2.6.2 Traffic type (TT)

The traffic type field is used to specify the data type for the home server to internally process packets when the Quality of Service (QoS) function is to be processed from the home server internal processing block.

The application program may configure the CCP header's traffic type by checking the APDU characteristics. The home server may or may not be designed to support QoS control according to the traffic type field value.

This document does not specify whether the home server should support QoS or the QoS support algorithm.

As shown in Table 2, the traffic type field consists of four bits.

**Table 2 – Traffic type field**

| Traffic type[3:0] | Notation           | Description                                                                                                       |
|-------------------|--------------------|-------------------------------------------------------------------------------------------------------------------|
| 0000              | Don't care         | Used when the traffic type does not need to be specified and any QoS processing is acceptable in the home server. |
| 0100              | Control Data       | Used when the APDU is a signaling message for managing or controlling HNMP, UHCP, HDSP or HMSP.                   |
| 1000              | Non Real-Time Data | Used for general data such as the Internet.                                                                       |
| 1100              | Real-Time Data     | Used for real-time data such as audio or video stream.                                                            |
| others            | reserved           | Reserved for future use.                                                                                          |

### 7.2.6.3 Payload type (PT)

The payload type field indicates the CCP payload type.

There are currently specifications regarding HNMP, UHCP, HDSP and HMSP defined in the CCP. In future, the payload type field can be used in protocols developed for other services using the CCP. As shown in Table 3, the payload type field consists of one byte.

**Table 3 – Payload type field**

| Payload type [7:0] | Notation            | Description                                                                   |
|--------------------|---------------------|-------------------------------------------------------------------------------|
| 0x00               | General application | Indicates that the payload type is part of the general application.           |
| 0x01               | HNMP                | Indicates that the payload type is an HNMP-related packet defined in the CCP. |
| 0x02               | UHCP                | Indicates that the payload type is an UHCP-related packet defined in the CCP. |
| 0x03               | HDSP                | Indicates that the payload type is an HDSP-related packet defined in the CCP. |
| 0x04               | HMSP                | Indicates that the payload type is an HMSP-related packet defined in the CCP. |
| others             | reserved            | Reserved for future use.                                                      |

### 7.2.7 Reserved (RSV) field

The five-byte reserved field is reserved for future use.

### 7.2.8 CCP payload length (CCPLEN) field

The four-byte CCP payload length field indicates the byte size of the APDUs in the payload.

### 7.2.9 CCP payload field

The CCP payload field comprises the APDUs generated by the application program. The CCP payload is written with the signaling messages of the HMSP, HNMP, UHCP and HDSP defined in the CCP.

## 8 Home network management protocol (HNMP)

### 8.1 General

The HNMP is used by the CCP to perform home network management functions. Moreover, the HNMP manages home network resources and provides PnP functions for the home server as well as the CCP devices.

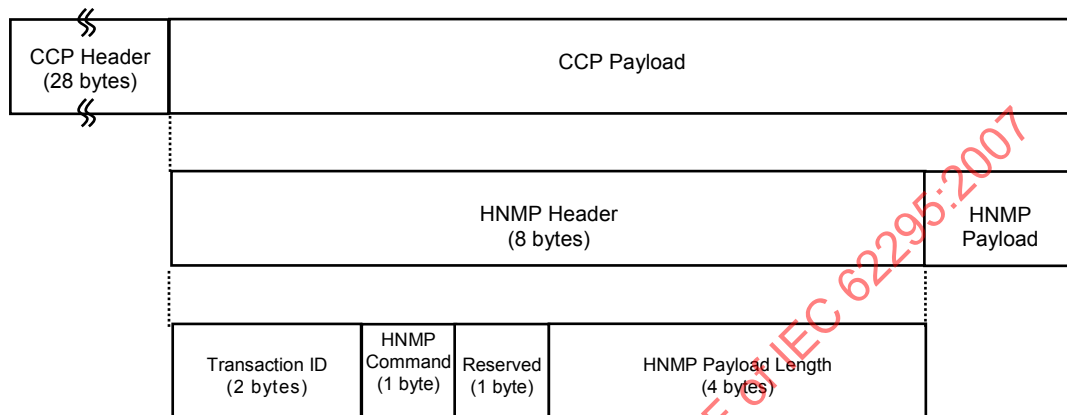
The HNMP is a basic packet-oriented signaling protocol provided by the CCP. Simple request and response packets are defined in the HNMP.

All CCP devices under the home network environment consisting of heterogeneous networks shall at least support the device registration function among the HNMP functions. Other HNMP functions can be selectively implemented according to the CCP devices' processing capability and application areas.

## 8.2 HNMP packet format

An HNMP packet comprises the CCP payload field. When the application program transmits an HNMP packet, it shall set the payload type of CCP header's type fields as 0x01 in order to notify that the CCP payload type is the HNMP packet.

Figure 11 displays the HNMP packet format. An HNMP packet consists of the eight-byte HNMP header and the HNMP payload. The HNMP header contains four fields: the Transaction ID, HNMP Command, Reserved and HNMP Payload Length.



IEC 2081/07

**Figure 11 – HNMP packet format**

### 8.2.1 Transaction ID (TID)

When an application program sends and receives multiple request and response packets, the transaction ID allows the application program to find the response packet for the corresponding request packet. The application program can assign random values for the transaction ID. The size of the transaction ID is two bytes.

### 8.2.2 HNMP command (HNMPCMD)

The HNMP command provides functions related to the home server registration, device registration, device management, alive-check, address information and traffic information. The HNMP command is one-byte long.

### 8.2.3 Reserved (RSV) field

The one-byte reservation field is reserved for future use.

### 8.2.4 HNMP payload length (HNMPPLEN) field

The HNMP payload length field indicates the byte size of the HNMP payload. HNMP payload length is a four-byte field.

### 8.2.5 HNMP payload

An HNMP command may or may not require an HNMP payload according to the HNMP command type.

DEV\_REG\_REQ, DEV\_REG\_RES, ADD\_DEV, DEL\_DEV and DEV\_INFO\_RES are examples of packets that require an HNMP payload.

### 8.3 Home server registration

The HNMP provides the home server registration function. Home server registration refers to the process of determining domain addresses among the home servers using the signaling packets of home servers that constitute multiple domain networks under the multi-HS network environment.

This document only considers the single-HS network environment and does not contain detailed information regarding home server registration.

### 8.4 Device registration

Device registration refers to the process of home server storing a human-readable name of a device and allocating a device ID used in the CCP addressing scheme to a CCP device. When a new CCP device is to be linked to a home network, it shall go through the device registration process. This allows the CCP device to have all CCP address fields such as Domain Address, Cluster Address and Device ID, through which the home server is able to notify other CCP devices that a new CCP device has been connected to the home network.

In order to perform the device registration process, all CCP devices within the home network shall be allocated with a unique address. Therefore, when carrying out the device registration process, the home server shall be able to allocate different device IDs to CCP devices with identical domain and cluster addresses.

Commands related to device registration are Device Registration Request (DEV\_REG\_REQ) and Device Registration Response (DEV\_REG\_RES) packets.

Figure 12 displays the packet format, and Figure 13 is an example command sequence related to device registration.

#### 8.4.1 Device registration request (DEV\_REG\_REQ) packet

Every CCP devices to be registered in a home network shall broadcast a DEV\_REG\_REQ packet to its cluster network. The broadcast packet is received by the home server HSI and delivered to the CCP layer. At this time, the CCP device shall fill the payload of a DEV\_REG\_REQ packet with its text-based name which is human-readable, physical address (or network address) size and physical address (or network address). As shown in Figure 12, support protocol field is used to indicate which protocols the CCP device additionally supports among UHCP, HDSP and HMSP. If the CCP device supports UHCP and HDSP, then the UHCP and HDSP fields should be set to 1 s.

The HSI shall notify the CCP device requesting registration with the Domain Address, Cluster Address, a newly allocated Device ID and its physical address (or network address). The HSI shall also register the CCP device's CCP address and physical address (or network address) in the address table for mutual communication between the CCP device and the HSI once the registration process is complete.

The reason why a device and a HSI exchange their physical addresses (or network addresses) is that both of them actually use physical addresses or network addresses when communicating with each other.

In case that the lower protocol layers of a device have both a network address and a physical address, it is easy to use network address rather than physical address when implementing an interface between the CCP layer and the lower protocol layers. On the other hand, in case the lower protocol layers of a device have only a physical address, there is no way but to use the physical address when interfacing between the CCP layer and the lower protocol layers.

The physical address or network address size is expressed in bytes. The HNMP command field of a DEV\_REG\_REQ packet is 0x31.

If there is no response for a timeout period after a DEV\_REG\_REQ packet is sent, the CCP device to be registered performs retries. If there is still no response, the home server is considered not ready and the procedure is repeated after a waiting period.

With respect to the timeout period, the number of retries and waiting periods are not specified in detail in this document because it is related to the communication speed influenced by the processing power of a device and the physical network speed.

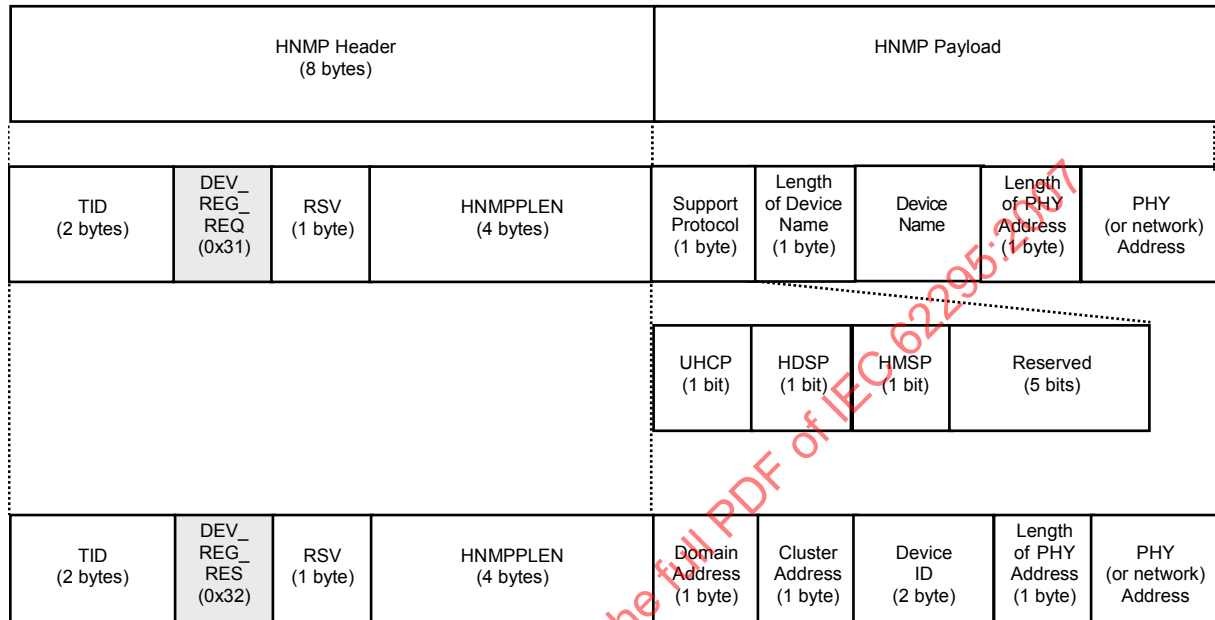


Figure 12 – DEV\_REG\_REQ and DEV\_REG\_RES packets

IEC 2082/07

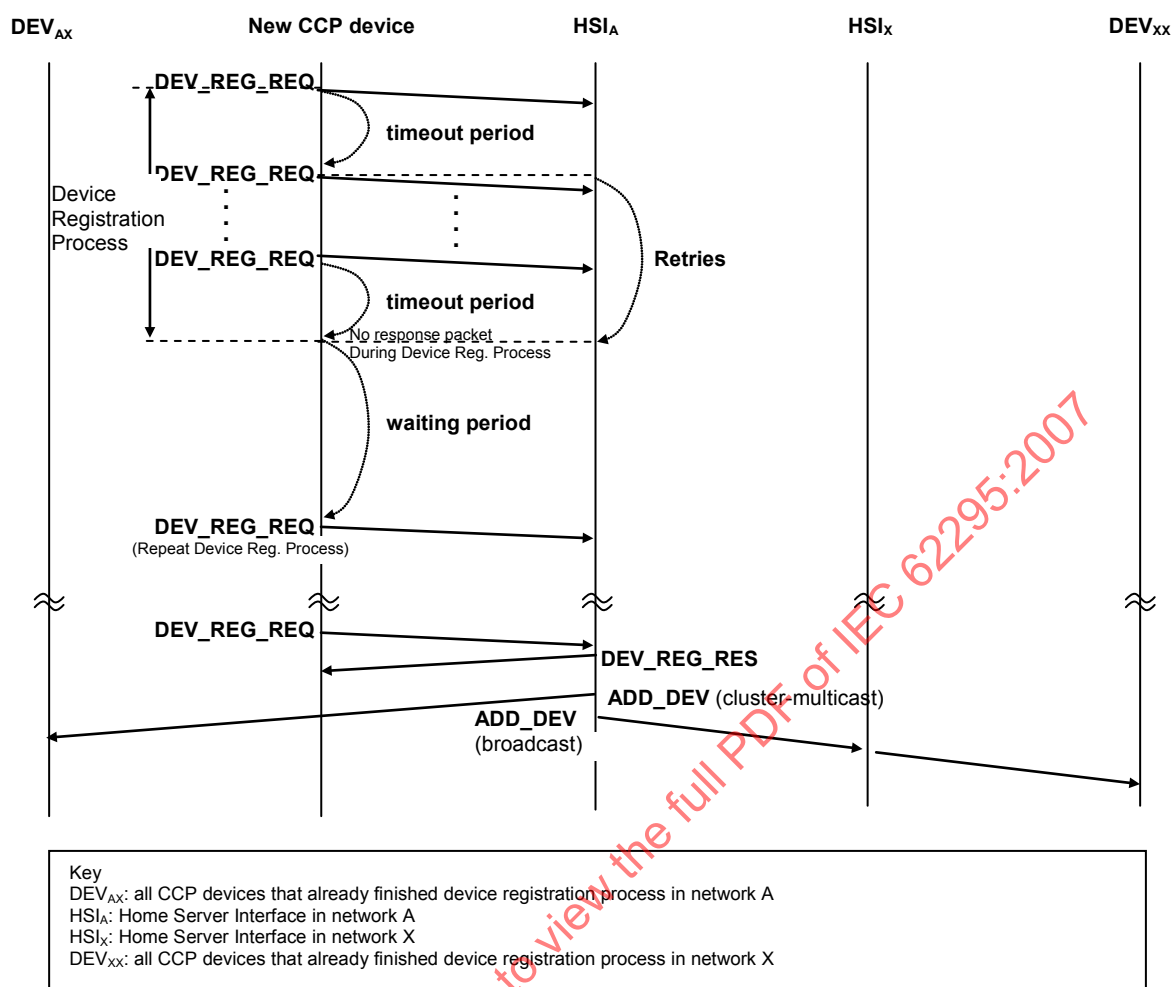
#### 8.4.2 Device registration response (DEV\_REG\_RES) packet

Upon receiving a DEV\_REG\_REQ packet, the HSI shall return a DEV\_REG\_RES packet to the DEV\_REG\_REQ sender device as a response within the timeout period. The HSI shall be able to store the physical address (or network address) of the device included in the DEV\_REG\_REQ packet payload by mapping it with the CCP address.

Furthermore, the HSI shall write its own domain address, cluster address, the device ID to be allocated to the device, physical address (or network address) size and the physical address (or network address) in the HNMP payload. The physical address (or network address) size is indicated in bytes. In order to send the DEV\_REG\_RES packet physically, the HSI shall use the CCP device's physical address (or network address) as the destination physical address (or network address).

After receiving the DEV\_REG\_RES packet, the CCP device configures its own domain address, cluster address and device ID and stores the physical address (or network address) of the HSI responsible for its cluster network. Henceforth, when the CCP device transmits data to another CCP device in a heterogeneous network, it simply has to configure the HSI's physical address (or network address) as the destination physical address (or network address) at the physical (or network) layer. Then, the HSI can receive the data at the physical layer and extracts a CCP packet at the CCP layer. Finally the HSI can forward the packet to another HSI to which the destination CCP device is linked.

The HNMP command field of the DEV\_REG\_RES packet is 0x32.



IEC 2083/07

**Figure 13 – Example of HNMP command sequence for device registration**

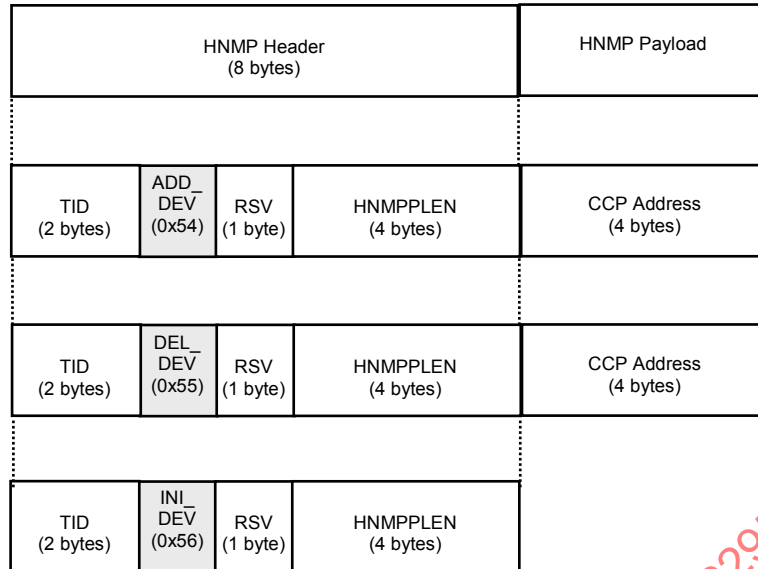
## 8.5 Device management

The HNMP provides HSIs and CCP devices with the address table management function through commands that allow addition and deletion of a device. It can also initialize a particular CCP device through the Device Initialization command.

Requirements for the HSIs (or CCP devices) receiving Add (or Delete) Device packets are as follows:

- The HSI shall be able to deliver the received Add (or Delete) Device packet to another cluster network's HSI.
- The HSI shall be able to deliver the Add (or Delete) Device packet from another HSI to all devices within its own cluster network.
- The HSI (or the CCP device) shall be able to add a new CCP address to the address table that it manages.
- The HSI (or CCP device) receiving a Delete Device packet shall be able to delete the address of the corresponding CCP device from the address table.
- Functions related to device addition and deletion shall be applied only to a limited number of HSIs and CCP devices with sufficient processing capabilities and memory.

The HNMP packet related to device addition, deletion and initialization consists of the Add Device (ADD\_DEV), Delete Device (DEL\_DEV) and Initialize Device (INI\_DEV) packets as shown in Figure 14.



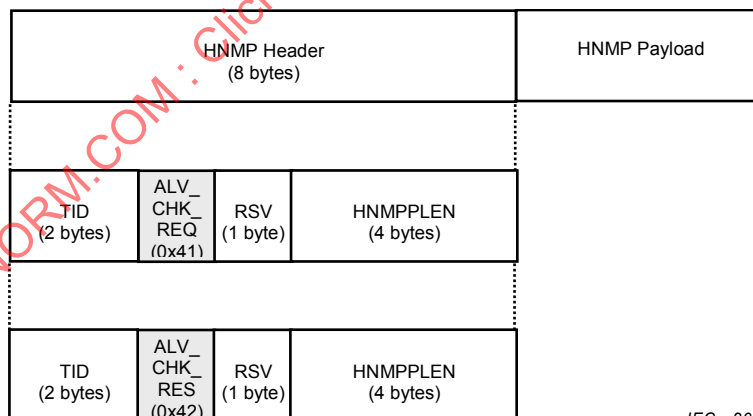
IEC 2084/07

**Figure 14 – ADD\_DEV, DEL\_DEV and INI\_DEV packets**

A CCP device (or HSI) can assess a particular CCP device's proper operation status and home network connection status by using HNMP's Alive-Check command.

The CCP device receiving an Alive-Check command shall be able to notify the fact that it is still connected to the home network and operating properly. In addition, in order to provide CCP devices' PnP functions, the HSI shall regularly check proper operation of the CCP devices within its cluster network using the Alive-Check command.

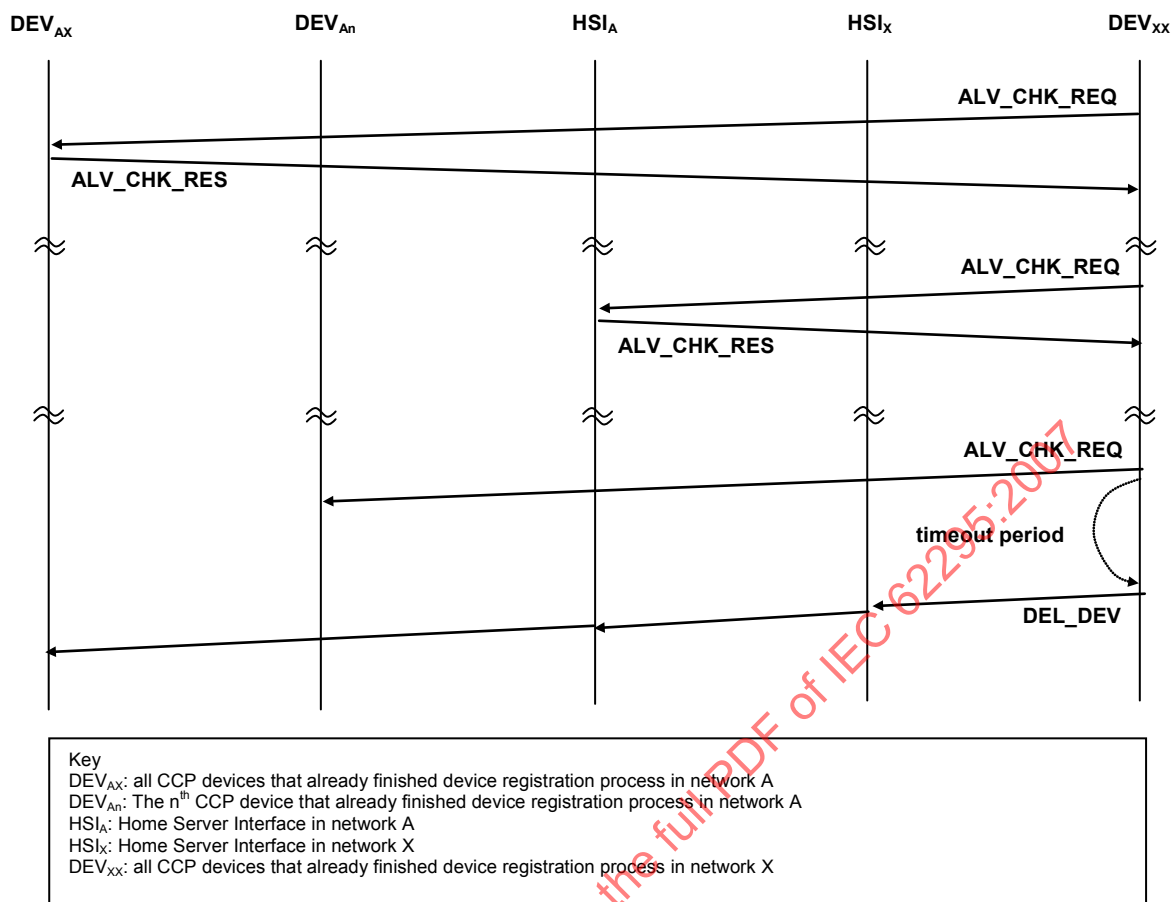
HNMP packets related to the Alive-Check command are Alive-Check Request (ALV\_CHK\_REQ) and Alive-Check Response (ALV\_CHK\_RES), as shown in Figure 15.



IEC 2085/07

**Figure 15 – ALV\_CHK\_REQ and ALV\_CHK\_RES packets**

Figure 16 displays an example of HNMP packet's command sequence related to device management.



IEC 2086/07

**Figure 16 – Example of HNMP command sequence for device management**

### 8.5.1 Add device (ADD\_DEV) packet

An ADD\_DEV packet is used to notify other CCP devices or HSIs of the fact that a new CCP device is connected to the home network. HSIs or CCP devices receiving the ADD\_DEV packet do not have to send a response packet.

Prior to transmission, the four-byte CCP address of the CCP device newly connected to the home network is provided in ADD\_DEV packet's HNMP payload. The ADD\_DEV packet is usually transmitted with the cast type of CCP header's type fields set as broadcast. A CCP device receiving the ADD\_DEV packet can process or discard it according to its address table management capability. Devices with address table management capability and a display panel shall be able to update the address table and indicate on the display panel that a new CCP device has been added. The HNMP command field of the ADD\_DEV packet is "0x54".

### 8.5.2 Delete device (DEL\_DEV) packet

A DEL\_DEV packet is used to notify other CCP devices and HSIs in the home network of the fact that a particular CCP device has been removed from the home network. For example, a CCP device (or HSI) transmits a DEL\_DEV packet if it has assessed that a particular CCP device no longer exists in the home network due to malfunctioning or other reasons. HSIs (or CCP devices) receiving a DEL\_DEV packet do not have to send a response packet. Prior to transmission, the four-byte CCP address of the CCP device removed from the home network is provided in DEL\_DEV packet's HNMP payload. The DEL\_DEV packet is usually transmitted with the cast type of CCP header's type fields set as broadcast. A CCP device receiving the DEL\_DEV packet can process or discard it according to its address table management

capability. Devices with address table management capability and a display panel shall be able to update the address table and indicate on the display panel that a particular CCP device has been removed. The HNMP command field of the DEL\_DEV packet is 0x55.

### 8.5.3 Initialize device (INI\_DEV) packet

An INI\_DEV packet can be sent from a CCP device or a HSI to another CCP device, and the CCP device receiving an INI\_DEV packet shall initialize itself. The initialized CCP device does not have to go through another device registration process. In other words, the domain address and the device ID shall be able to maintain their values after initialization. The HNMP command field of the INI\_DEV packet is 0x56.

### 8.5.4 Alive-check request (ALV\_CHK\_REQ) packet

A CCP device (or home server) can transmit an ALV\_CHK\_REQ packet to another CCP device (or home server). An ALV\_CHK\_REQ packet is used to check whether a particular device is connected to the home network and operating properly. A CCP device or home server receiving an ALV\_CHK\_REQ packet shall return a response packet within two seconds. In order to support devices' PnP functions, a HSI shall regularly send ALV\_CHK\_REQ packets to devices within its cluster network and receive responses. If a HSI does not receive a response from a CCP device for a timeout period after sending an ALV\_CHK\_REQ packet, it retries the process  $N$  times. If there is no response after  $N$  retries, the corresponding CCP device shall be regarded as having been removed from the home network, and the HSI sends a DEL\_DEV packet to other HSIs and all CCP devices within the cluster network. This regulation does not specify the number of retries ( $N$ ). The HNMP command field of the ALV\_CHK\_REQ packet is 0x41.

### 8.5.5 Alive-check response (ALV\_CHK\_RES) packet

A CCP device (or HSI) that receives an ALV\_CHK\_REQ packet shall return an ALV\_CHK\_RES packet to the device where the ALV\_CHK\_REQ packet originated within a timeout period to notify that it is connected to the home network and operating properly. The HNMP command field of the ALV\_CHECK\_RES packet is 0x42.

## 8.6 Address and name information of devices

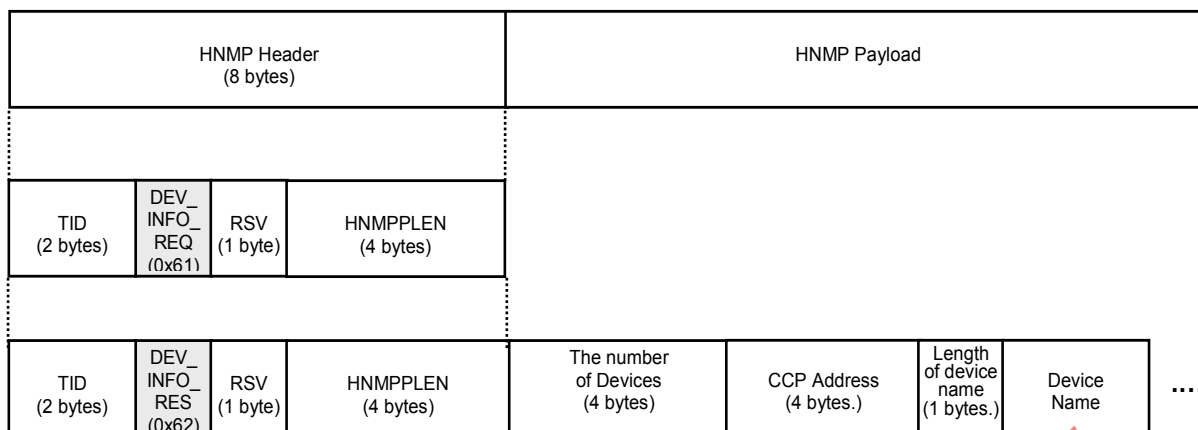
When a CCP device requests a HSI for CCP address and name information for all CCP devices linked to the home network, the HNMP supports the function of notifying the CCP address and name information of all CCP devices. This function can be used by a newly connected CCP device to request the CCP address information of existing CCP devices or a CCP device to update address information of other CCP devices that it manages.

As shown in Figure 17, there are two types of HNMP packets related to the address and name information: Device Information Request (DEV\_INFO\_REQ) and Device Information Response (DEV\_INFO\_RES).

A DEV\_INFO\_REQ packet can be transmitted in one of two ways.

First, if a particular CCP device transmits a DEV\_INFO\_REQ packet to a HSI as unicast, it can obtain address information of all CCP devices within its own cluster network.

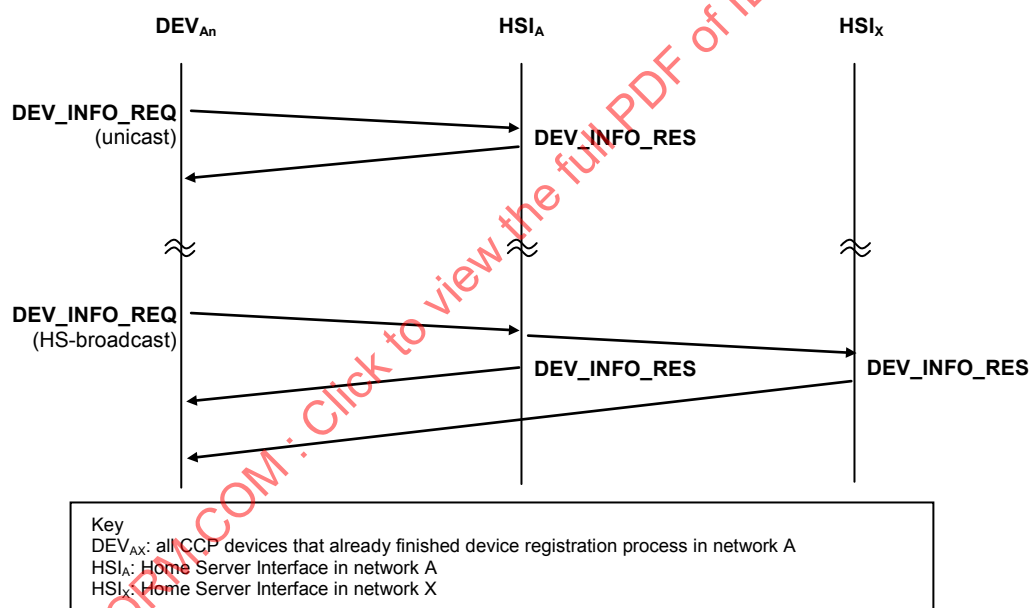
Second, if a particular CCP device transmits a DEV\_INFO\_REQ packet as HS-broadcast, it can obtain address information of all CCP devices in all cluster networks within the home network.



IEC 2087/07

Figure 17 – DEV\_INFO\_REQ and DEV\_INFO\_RES packets

Figure 18 displays an example command sequence of the HNMP packet related to the device address and name information.



IEC 2088/07

Figure 18 – Example of HNMP command sequence for retrieving device address and name information

### 8.6.1 Device address and name information request (DEV\_INFO\_REQ) packet

A DEV\_INFO\_REQ packet is used by a CCP device with address table management capability to request a HSI for address and name information of all CCP devices linked to the cluster network.

A HSI receiving a DEV\_INFO\_REQ packet shall write the CCP address and name information of all CCP devices registered in its cluster network in the HNMP payload of the response packet and return the response packet to the CCP device that has sent the request packet. The HNMP command field of the DEV\_INFO\_REQ packet is 0x61.

### 8.6.2 Device address and name information response (DEV\_INFO\_RES) packet

A HSI receiving a DEV\_INFO\_REQ packet shall return a DEV\_INFO\_RES packet to the CCP device that has requested the address and name information of CCP devices linked to itself.

In Figure 17, “The number of Devices” field shall be four byte long and shall contain the number of CCP device information listed in the payload of DEV\_INFO\_RES packet. CCP address, length of device name and device name fields represent information of a specific CCP device. The CCP address field shall be four byte long and contain a CCP address of a specific CCP device. Length of device name field shall be one byte long and contain the length of a device name in byte units. The device name field shall contain the name of a specific CCP device in simple text format. Its size shall be equal to the length of device name field.

A DEV\_INFO\_RES packet can be transmitted in one of two ways.

First, if a HSI received a DEV\_INFO\_REQ packet has  $N$  devices linked to itself, then the HSI makes a DEV\_INFO\_RES packet of which the number of devices field is set to 1, and it sends it  $N$  times for each device.

Second, a HSI that received a DEV\_INFO\_REQ packet makes only one DEV\_INFO\_RES packet which have address and name information of all CCP devices linked to itself.

The HNMP command field of the DEV\_INFO\_RES packet is 0x62.

## 8.7 Other management functions

In addition to the functions specified above, new management packets can be defined and applied when a home network is expanded or new functions become necessary.

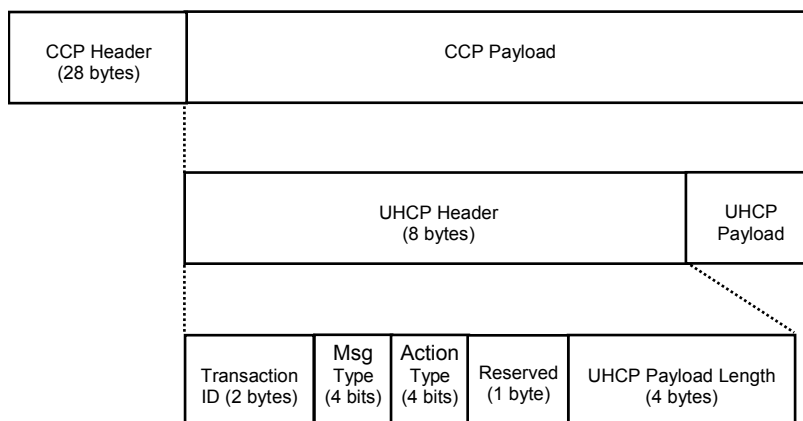
## 9 Universal home control protocol (UHCP)

CCP provides UHCP for device control and monitoring functions. UHCP is a message-based protocol that can provide home network resource controlling and monitoring functions. One of the fundamental protocols provided by CCP, UHCP is used for additional services offered by devices with CCP capabilities.

Since basic home network management is performed by HNMP, not all devices need to be embedded with UHCP. However, if UHCP is implemented according to CCP device processing capability, it can provide additional home network services such as control and monitoring.

### 9.1 UHCP packet format

Figure 19 displays the UHCP packet format. The UHCP packet is located in the CCP packet payload. In order to indicate that the CCP payload type is a UHCP packet, the UHCP application program shall set the payload type of the CCP header's type fields as 0x02. An UHCP packet consists of an eight-byte UHCP header and the UHCP payload. The UHCP header contains five fields: Transaction ID, message type, action type, reserved and UHCP payload length.



IEC 2089/07

**Figure 19 – UHCP packet format****9.1.1 Transaction ID (TID)**

When an application program sends and receives multiple request packets and response packets, the transaction ID is used by the application program to find a response packet for a particular request packet. The application program can assign random values for the transaction ID. The transaction ID is a two-byte field.

**9.1.2 Message type (MT) and action type (AT)**

The message type field is four bits long and it consists of the execution message, query message and notification message. The action type field is four bits long and it indicates detailed actions according to the message type.

Figure 20 displays the message type and action type defined by the UHCP.

| UHCP<br>(8 bytes) |               |                       |     |          | UHCP Payload |
|-------------------|---------------|-----------------------|-----|----------|--------------|
|                   |               |                       |     |          |              |
| TID               | EXE<br>(0x1)  | REG<br>(0x1)          | RSV | UHCPPLEN | UHCP Payload |
| TID               | EXE<br>(0x1)  | CTRL<br>(0x2)         | RSV | UHCPPLEN | UHCP Payload |
| TID               | EXE<br>(0x1)  | RES<br>OK<br>(0xE)    | RSV | UHCPPLEN |              |
| TID               | EXE<br>(0x1)  | RES<br>NOK<br>(0xF)   | RSV | UHCPPLEN |              |
| TID               | QUE<br>(0x2)  | REG<br>STAT<br>(0x1)  | RSV | UHCPPLEN |              |
| TID               | QUE<br>(0x2)  | CTRL<br>STAT<br>(0x2) | RSV | UHCPPLEN |              |
| TID               | QUE<br>(0x2)  | ALL<br>STAT<br>(0x3)  | RSV | UHCPPLEN |              |
| TID               | QUE<br>(0x2)  | RES<br>OK<br>(0xE)    | RSV | UHCPPLEN | UHCP Payload |
| TID               | QUE<br>(0x2)  | RES<br>NOK<br>(0xF)   | RSV | UHCPPLEN |              |
| TID               | NTFY<br>(0x3) | N/A<br>(0x0)          | RSV | UHCPPLEN | UHCP Payload |

IEC 2090/07

**Figure 20 – Message type and action type fields of UHCP packet**

### 9.1.3 Reserved (RSV) field

The one-byte reserved field is for future use.

### 9.1.4 UHCP payload length (UHCPPLEN) field

The UHCP payload length field indicates the byte size of the UHCP payload. The size of the length field is four bytes.

### 9.1.5 UHCP payload

The UHCP payload may or may not be required according to the message type and the action type.

## 9.2 Execution messages (EXE)

If the UHCP header's MT field is 0x1, it indicates that the content of the UHCP payload is an execution message. There are four action types according to the execution message: EXE\_REG, EXE\_CTRL, EXE\_RESOK and EXE\_RESNOK.

### 9.2.1 Execution of registration (EXE\_REG)

The AT field of an EXE\_REG message is 0x1. In order for an UHCP-supporting device to use UHCP functions after completing the HNMP device registration process, it shall first register

its control items and device attributes such as the device type, location, manufacturer and network technology, to the HSI in the domain network. The EXE\_REG message is used for registering this kind of information. The device attributes and control items are indicated in the payload.

As shown in Figure 21, a HSI receiving an EXE\_REG message shall send a response message within the timeout period. If the device that has sent the EXE\_REG message does not receive a response within the timeout period, it carries out retries. If it fails to receive a response after retries of EXE\_REG, the home server is regarded as problematic, and the process is repeated after thirty seconds. Since the device must register its attributes through the process in order to perform UHCP functions, if there is no response from the HSI, it continuously repeats the process until registration is complete.

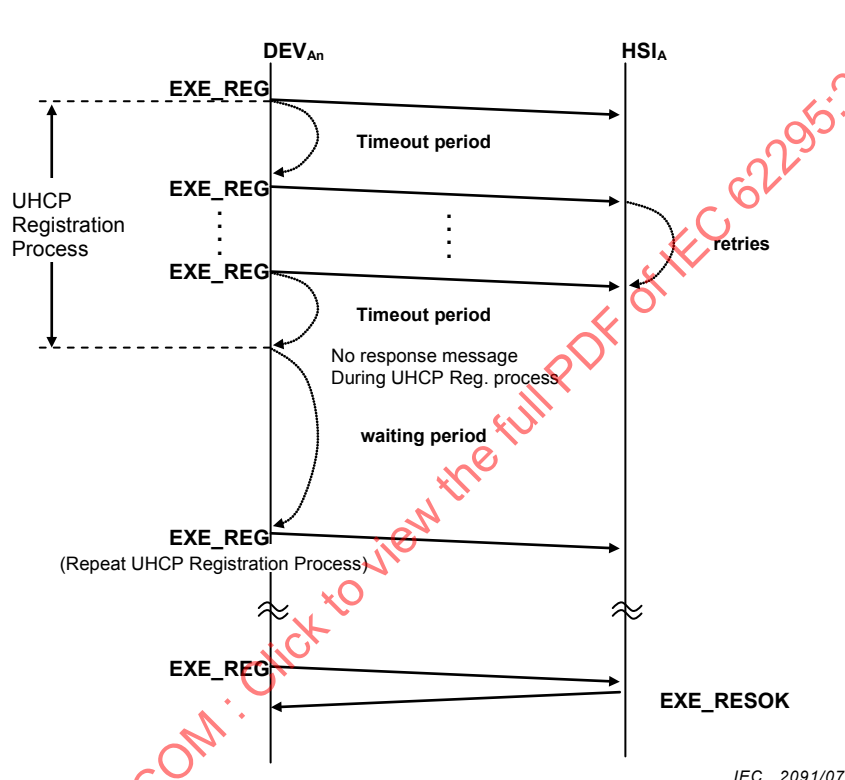


Figure 21 – Example of registration process

### 9.2.2 Execution of control (EXE\_CTRL)

The AT field of an EXE\_CTRL message is 0x2. A device receiving an EXE\_CTRL packet shall carry out the command according to the payload content and send a corresponding response message.

Figure 22 displays the sequence of an EXE\_CTRL message. If the CCP device that has sent an EXE\_CTRL message does not receive an EXE\_RESOK response message within timeout period, it sends an EXE\_RESNOK message to the UHCP application program. Even if the UHCP application program does not receive a response packet for an EXE\_CTRL message, it does not attempt a retry. However, an application program using UHCP functions can decide the timeout period and perform a retry according to a user request.

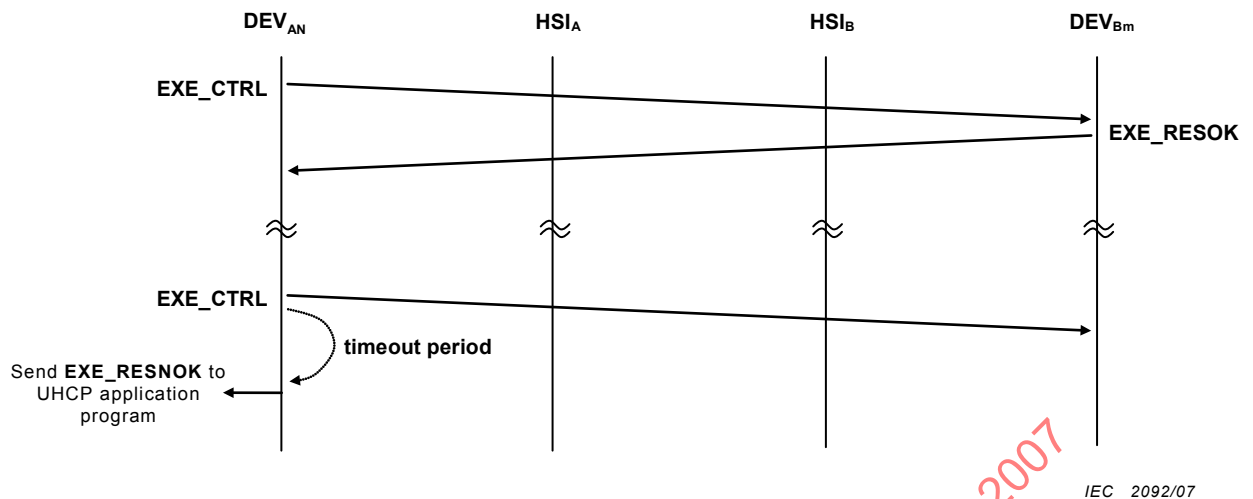


Figure 22 – Example of EXE\_CTRL message

### 9.2.3 Response OK (EXE\_RESOK)

The AT field of an EXE\_RESOK message is 0xE. A device receiving an EXE\_REG or EXE\_CTRL packet shall complete the device registration or carry out a command and return an EXE\_RESOK message indicating that the task was performed successfully. An EXE\_RESOK message does not have an UHCP payload. (Refer to Figure 20)

### 9.2.4 Response NOK (EXE\_RESNOK)

The AT field of an EXE\_RESNOK message is 0xF. A device receiving an EXE\_REG or EXE\_CTRL packet shall return an EXE\_RESNOK message to indicate that the registration process or a command has failed. In case of the EXE\_REG message, if the device that has sent an EXE\_REG message fails to receive an EXE\_RESOK or EXE\_RESNOK response due to an unstable home server, the device shall be able to send an EXE\_RESNOK message to the UHCP application on its own. An EXE\_RESNOK message does not have an UHCP payload as shown in Figure 20.

## 9.3 Query messages (QUE)

If the MT field of an UHCP header is 0x2, it indicates that the UHCP payload content is a query or monitoring message. There are five types of execution messages in the action type according to the query message: QUE\_REGSTAT, QUE\_CTRLSTAT, QUE\_ALL, QUE\_RESOK and QUE\_RESNOK, as described below.

### 9.3.1 Query of registration status (QUE\_REGSTAT)

The AT field of a QUE\_REGSTAT message is 0x1. It is used by a home server or another device to request for a particular device's attributes. A device receiving a QUE\_REGSTAT message shall provide its attribute information in QUE\_RESOK message's UHCP payload and return it. If there is no response within the timeout period, the device that has sent the QUE\_REGSTAT message shall send a QUE\_RESNOK message to the UHCP application program. Figure 23 displays an example of a QUE\_REGSTAT message. A QUE\_REGSTAT message does not have an UHCP payload, as shown in Figure 20.

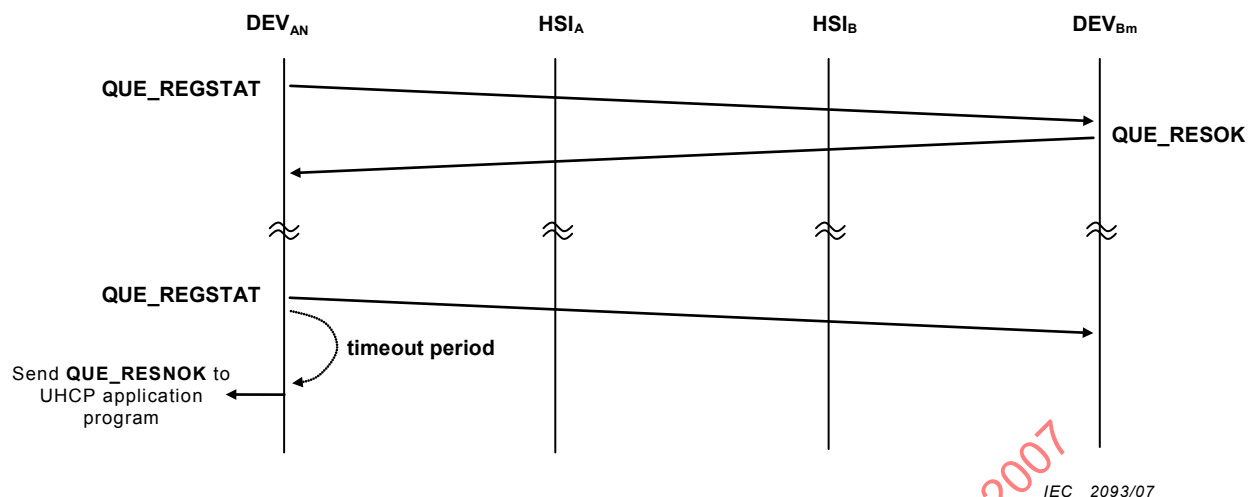


Figure 23 – Example of QUE\_REGSTAT message

### 9.3.2 Query of control status (QUE\_CTRLSTAT)

The AT field of a **QUE\_CTRLSTAT** message is 0x2. It is used by a home server or another device to request for information regarding a particular device's command status. A device receiving a **QUE\_CTRLSTAT** message shall provide its command status information in **QUE\_RESOK** message's UHCP payload and return it. If there is no response within timeout period, the device that has sent the **QUE\_CTRLSTAT** message shall send a **QUE\_RESNOK** message to the application program. Figure 24 displays an example of a **QUE\_CTRLSTAT** message. A **QUE\_CTRLSTAT** message does not have an UHCP payload as shown in Figure 20.

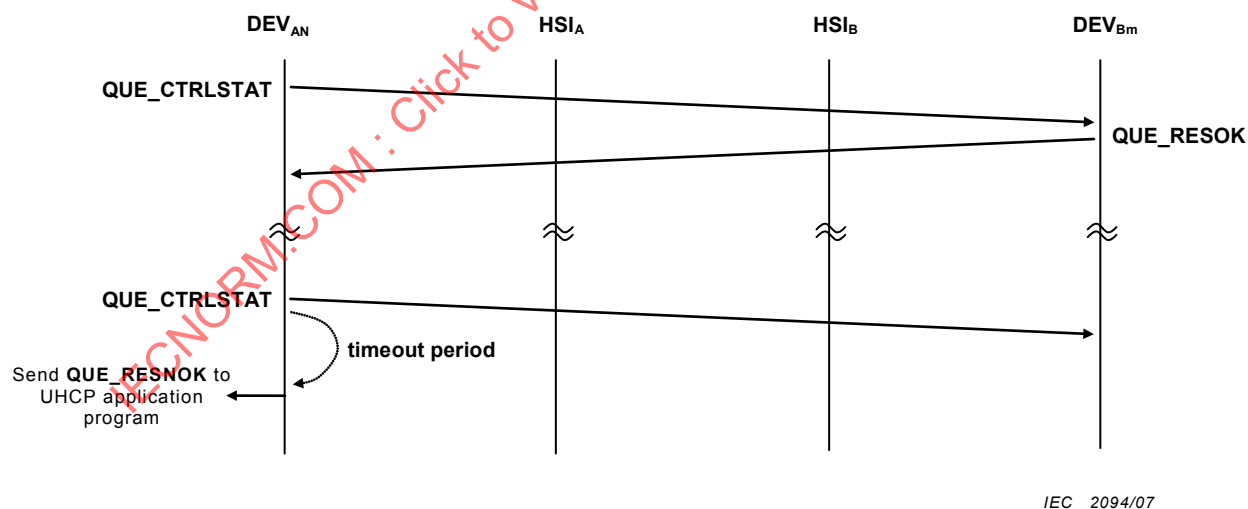


Figure 24 – Example of QUE\_CTRLSTAT message

### 9.3.3 Query of all status (QUE\_ALLSTAT)

The AT field of a **QUE\_ALLSTAT** message is 0x3. It is used by a home server or another device to request for a particular device's command status as well as its attributes. A device receiving a **QUE\_ALLSTAT** message shall provide its command status and attribute information in **QUE\_RESOK** message's UHCP payload and return it. If there is no response within a timeout period, the device that has sent the **QUE\_ALLSTAT** message shall send a **QUE\_RESNOK** message to the application program. A **QUE\_ALLSTAT** message does not have an UHCP payload, as shown in Figure 20.

Figure 25 displays an example of a QUE\_ALLSTAT message.

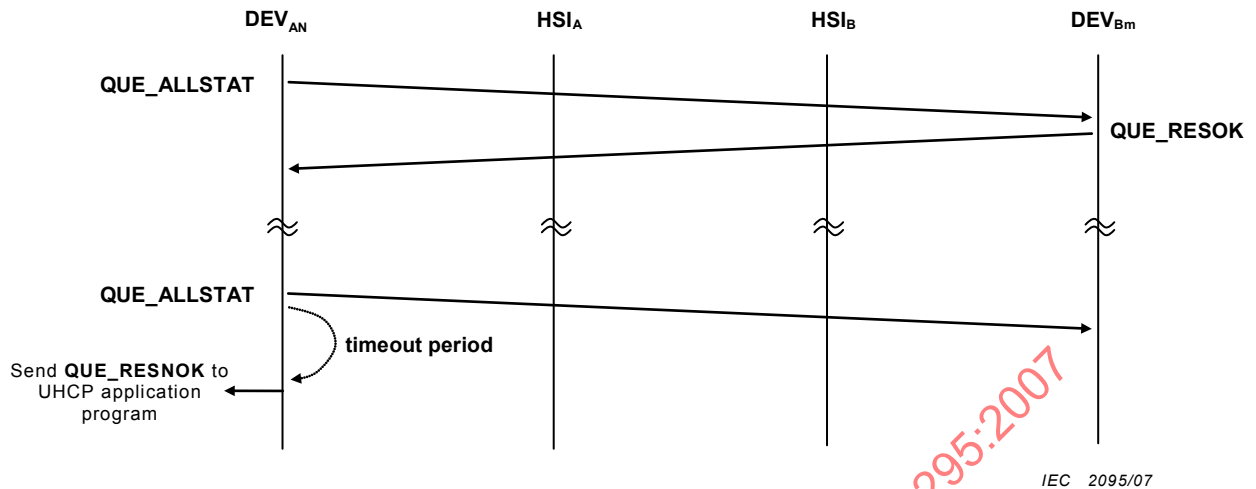


Figure 25 – Example of QUE\_ALLSTAT message

#### 9.3.4 Response OK (QUE\_RESOK)

The AT field of a QUE\_RESOK message is 0xE. It is used by a device receiving a QUE\_REGSTAT, QUE\_CTRLSTAT or QUE\_ALLSTAT message to notify its command status and attributes, information which is written in the UHCP payload for transmission.

#### 9.3.5 Response NOK (QUE\_RESNOK)

The AT field of a QUE\_RESNOK message is 0xF. If a device receiving a QUE\_REGSTAT, QUE\_CTRLSTAT or QUE\_ALLSTAT message fails to query information, it shall send a QUE\_RESNOK message. In addition, if a device that has sent a query message cannot receive a response message due to malfunctioning in the device that has received the query message, the query-requesting device shall be able to send a QUE\_RESNOK message to the UHCP application program on its own. A QUE\_RESNOK message does not have an UHCP payload. (Refer to Figure 20.)

### 9.4 Notification messages (NTFY)

If the MT field of an UHCP header is 0x3, it indicates that the UHCP payload content is a notification message. A notification message is used to notify the HSI of abnormal operation such as device malfunctioning or unstable power supply, as well as cases where specific circumstances shall be communicated to the user. An action type is not defined in an NTFY message, and the content of notification is written in the UHCP payload.

### 9.5 UHCP payload syntax

The syntax used in the UHCP payload is similar to that of markup language and shall meet the following conditions.

#### 9.5.1 Basic syntax for UHCP payload

The content of the UHCP payload shall be created according to the following six fundamental rules.

It shall be written only in text (upper and lower case letters), numbers, "<", ">" and "/".

- a) Terms indicated in upper case letters are items used in the UHCP payload and they are reserved words. Reserved words shall not be used as arguments.
- b) The character “/” shall not be used in an argument.
- c) Italicized terms are not reserved words used in the UHCP payload, and they can be specified according to the user’s discretion.
- d) UHCP payload content is expressed as “<ITEM>*argument*</ITEM>”, which means that the *item* shall have the value of the *argument* or take actions specified by the *argument*.
- e) UHCP payload shall begin with a “<UHCP>” and end with a “</UHCP>”.
- f) Each item within “<>” shall be indicated in upper case letters.

### 9.5.2 Syntax for UHCP registration

For a device to register its basic attributes and control items, an UHCP registration process is required. The UHCP payload syntax and reserved words for this purpose are specified as below.

- a) Indication of registering device attributes and control items shall begin with a <REG> and end with a </REG>.
- b) Indication of device attributes shall begin with a <ATTR> and end with a </ATTR>.
- c) Device attributes shall consist of four items: name of device, name of vendor, location of device and type of network used by device.
- d) Name of the device shall be written as “<DEV> *name\_of\_device*</DEV>”.
- e) Name of the vendor who has manufactured the device shall be written as “<VEN>*name\_of\_vendor*</VEN>”.
- f) Location of the device shall be written as “<LOC> *location\_of\_device*</LOC>”.
- g) Type of network protocol used by the device shall be written as “<NET>*name\_of\_network\_protocol*</NET>”.
- h) Indication of device’s control items shall begin with a <CMD> and end with a </CMD>.
- i) One or more control items shall be written between <CMD> and </CMD> in the format of “<COMMAND\_TYPE>*current\_status*</COMMAND\_TYPE>”. “<COMMAND\_TYPE>” is not a reserved word, and common terms should be used to facilitate understanding of the user.
- j) Indication of monitoring items shall start with a <MON> and end with a </MON>.
- k) One or more monitoring items shall be written between <MON> and </MON>. A meter shall be written as “<METER>*value*</METER>”, and a sensor as “<SENSOR>*value*</SENSOR>”.

(Example 1) UHCP payload content for UHCP registration of a DVD player:

```
<UHCP>
  <REG>
    <ATTR>
      <DEV>DVD Player</DEV>
      <VEN>vendor</VEN>
      <LOC>Living Room</LOC>
      <NET>IPv4</NET>
    </ATTR>
    <CMD>
      <PLAY>off</PLAY>
      <PAUSE>off</PAUSE>
      <REW>off</REW>
      <FF>off</FF>
    </CMD>
  </REG>
</UHCP>
```

(Example 2) UHCP payload content for UHCP registration of an electric meter:

```
<UHCP>
  <REG>
    <ATTR>
      <DEV>electric meter</DEV>
      <VEN>vendor</VEN>
      <LOC>Living Room</LOC>
      <NET>PLC</NET>
    </ATTR>
    <MON>
      <METER>100kwh</METER>
    </MON>
  </REG>
</UHCP>
```

### 9.5.3 Syntax for device control

The following reserved words and syntax are specified to express device control commands in the UHCP payload.

- Shall be used for EXE\_CTRL messages.
- A control command shall begin with a <CTRL> and end with a </CTRL>.
- One or more control items shall be written between <CMD> and </CMD> in the format of “<COMMAND\_TYPE>current\_status</COMMAND\_TYPE>”. “<COMMAND\_TYPE>” is not a reserved word, and common terms should be used to facilitate understanding of the user.

(Example 1) Payload content of an EXE\_CTRL message for controlling a camcorder:

```
<UHCP>
  <CTRL>
    <CMD>
      <PLAY>on</PLAY>
    </CMD>
  </CTRL>
</UHCP>
```

(Example 2) Payload content of an EXE\_CTRL message for controlling a dimming light:

```
<UHCP>
  <CTRL>
    <CMD>
      <POWER>on</POWER>
      <DIM_LEVEL>1</DIM_LEVEL>
    </CMD>
  </CTRL>
</UHCP>
```

#### 9.5.4 Syntax for query of controlling and monitoring status

The following reserved words and syntax are specified to express in the UHCP payload the commands for querying a device's UHCP registration status, control status or monitoring information.

- a) Shall be used for QUE\_RESOK messages that are responses to query messages such as QUE\_REGSTAT, QUE\_CTRLSTAT and QUE\_ALLSTAT.
- b) A response to a query message shall begin with a <STAT> and end with a </STAT>.
- c) A response to a QUE\_REGSTAT message shall begin with a <ATTR> and end with a </ATTR>. Device attributes shall be written between <ATTR> and </ATTR>.
- d) Means of indicating device attributes are identical to 10.5.2.
- e) A response to a QUE\_CTRLSTAT shall begin with a <CMD> and end with a </CMD>. Indication of monitored values, for these cases as a meter or sensor, it shall begin with a <MON> and end with a </MON>. In case of a meter, the monitored value shall be written as "<METER>value</METER>" between <MON> and </MON>. In case of a sensor, the sensed value shall be written as "<SENSOR>value</SENSOR>" between <MON> and </MON>.
- f) Means of indicating a general control status are identical to 10.5.3.

(Example 1) Payload content of a TV set's QUE\_RESOK message for a QUE\_REGSTAT message:

```
<UHCP>
  <STAT>
    <ATTR>
      <DEV>TV</DEV>
      <VEN>vendor</VEN>
      <LOC>Living Room</LOC>
      <NET>IEEE1394</NET>
    </ATTR>
  </STAT>
</UHCP>
```

(Example 2) Payload content of an electric meter's QUE\_RESOK message for a QUE\_CTRLSTAT message:

```
<UHCP>
  <STAT>
    <MON>
      <METER>100kwh</METER>
    </MON>
  </STAT>
</UHCP>
```

(Example 3) Payload content of a QUE\_RESOK message for a QUE\_ALLSTAT message:

```
<UHCP>
  <STAT>
    <ATTR>
      <DEV>TV</DEV>
      <VEN>vendor</VEN>
      <LOC>Living Room</LOC>
      <NET>IEEE1394</NET>
    </ATTR>
    <CMD>
      <POWER>on</POWER>
      <CHANNEL>11</CHANNEL>
      <VOLUME>20</VOLUME>
    </CMD>
  </STAT>
</UHCP>
```

### 9.5.5 Syntax for notification

- Automatic notification shall start with a <NOTIFY> and end with a </NOTIFY>.
- Shall be used for a device's automatic notification (NTFY) and written in the format of "<NOTIFY>*alarm\_contents*</NOTIFY>".

(Example) In case of a digital camera:

<UHCP>

<NOTIFY>Low Battery</NOTIFY>

<NOTIFY>No memory card</NOTIFY>

</UHCP>

## 10 Home data service protocol (HDSP)

CCP provides the HDSP for implementing data services such as file and directory services under the home network environment.

HDSP is a packet-based signaling protocol, one of the fundamental protocols provided by the CCP. HDSP defines simple request/response packets and performs file management and transmission functions within the home network.

All CCP devices that support data services under the home network environment constructed with heterogeneous networks shall support the HDSP.

This document only specifies HDSP's functional requirements.

### 10.1 Functional requirements of HDSP

Devices and home servers that provide HDSP shall satisfy the following requirements.

#### 10.1.1 Interoperability with CCP

In order to be compatible with CCP, HDSP shall satisfy the following requirements.

- HDSP shall be an application program using the CCP. Namely, commands or action results defined in relation with the HDSP shall be written in the CCP payload.
- The payload type field value in CCP header's type fields shall be 0x3.

#### 10.1.2 File and directory services

HDSP shall provide functions that allow the user to access the files dispersed throughout various devices in the home network. The files shall be readily available to the user regardless of time and place, as if they were stored in a single device.

In order to provide this kind of user-friendliness, HDSP shall satisfy the following requirements.

- HDSP shall provide functions for a particular device to request for information regarding files and directories stored in other devices.
- HDSP shall provide functions for a device receiving a file or directory information request to respond back with its corresponding information.
- HDSP shall provide functions for a particular device to copy, move, delete and change names of files and directories stored in other devices.
- HDSP shall provide functions for a particular device to request transfers of files and directories stored in other devices.
- HDSP shall provide functions for a device receiving a file or directory transfer request to respond back with a corresponding transfer to the requesting device.
- The device requesting a file or directory transfer shall be able to save the incoming file or device in its storage.

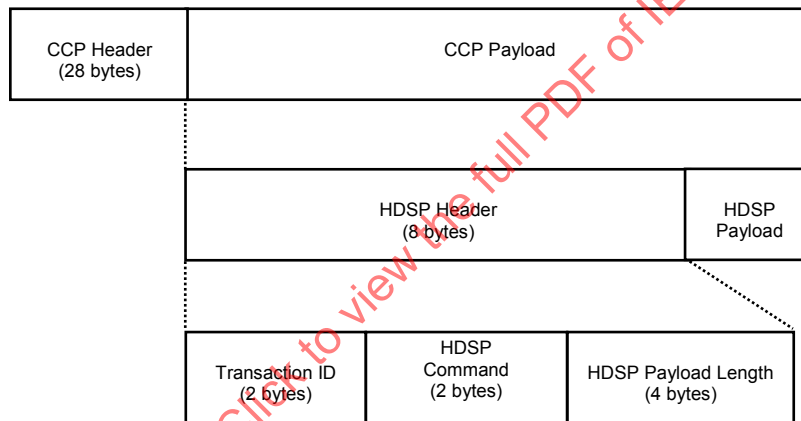
### 10.1.3 Messaging service

HDSP shall provide functions for the devices offering data services within the home network to exchange text messages. In order to do so, HDSP shall satisfy the following requirements.

- HDSP shall provide functions for a particular device to send a real-time message to another device.
- HDSP shall provide a reservation function for a particular device to send a text message to another device at a specified time. In order to do so, the home server shall be able to temporarily store the messages to be exchanged between two devices and transmit them at the specified time.
- A device receiving a message shall be able to display the message on a display screen.

### 10.2 HDSP packet format

Figure 26 represents the HDSP packet format. The HDSP packet is located in the CCP packet payload. In order to indicate that the CCP payload type is a HDSP packet, the HDSP application program shall set the payload type of the CCP header's option field as 0x03. A HDSP packet consists of an eight-byte HDSP header and the HDSP payload. The HDSP header contains three fields: Transaction ID, HDSP command, and HDSP payload length.



IEC 2096/07

**Figure 26 – HDSP packet format**

#### 10.2.1 Transaction ID (TID)

When an application program sends and receives multiple request packets and response packets, the transaction ID is used by the application program to find a response packet for a particular request packet. The application program can assign random values for the transaction ID. The transaction ID is a two-byte field.

#### 10.2.2 HDSP command

The HDSP command field is two bytes long. Detailed HDSP commands are shown in Table 4.

**Table 4 – HDSP commands**

| HDSP commands | HEX    | Description                                             |
|---------------|--------|---------------------------------------------------------|
| DIR_QUE_REQ   | 0x1011 | Request packet for retrieving directory information.    |
| DIR_QUE_RES   | 0x1012 | Response packet corresponding to DIR_QUE_REQ.           |
| DIR_DEL_REQ   | 0x1021 | Request packet for deletion of a directory.             |
| DIR_DEL_RES   | 0x1022 | Response packet corresponding to DIR_DEL_REQ.           |
| DIR_REN_REQ   | 0x1031 | Request packet for renaming of a directory.             |
| DIR_REN_RES   | 0x1032 | Response packet corresponding to DIR_REN_REQ.           |
| DIR_MAKE_REQ  | 0x1041 | Request packet for making of a directory.               |
| DIR_MAKE_RES  | 0x1042 | Response packet corresponding to DIR_MAKE_REQ.          |
| FILE_QUE_REQ  | 0x2011 | Request packet for an inquiring about file information. |
| FILE_QUE_RES  | 0x2012 | Response packet corresponding to FILE_QUE_REQ.          |
| FILE_DEL_REQ  | 0x2021 | Request packet for deletion of a file.                  |
| FILE_DEL_RES  | 0x2022 | Response packet corresponding to FILE_DEL_REQ.          |
| FILE_REN_REQ  | 0x2031 | Request packet for renaming of a file.                  |
| FILE_REN_RES  | 0x2032 | Response packet corresponding to FILE_REN_REQ.          |
| FILE_NEGO_REQ | 0x2051 | Request packet for negotiating for transferring a file. |
| FILE_NEGO_RES | 0x2052 | Response packet corresponding to FILE_NEGO_REQ.         |
| FILE_GET_REQ  | 0x2061 | Request packet for copying a file from a remote device. |
| FILE_GET_RES  | 0x2062 | Response packet corresponding to FILE_GET_REQ.          |
| FILE_PUT_REQ  | 0x2071 | Request packet for copying a file to a remote device.   |
| FILE_PUT_RES  | 0x2072 | Response packet corresponding to FILE_PUT_REQ.          |
| MSG_PUT_REQ   | 0x3071 | Request packet for sending a text message.              |
| MSG_PUT_RES   | 0x3072 | Response packet corresponding to MSG_SEND_REQ.          |

### 10.2.3 HDSP payload length (HDSPPLEN) field

The HDSP payload length field indicates the byte size of the HDSP payload. The size of the length field is four bytes.

### 10.2.4 HDSP payload

The HDSP payload contains information required for a HDSP command.

## 10.3 Messages for directory services

HDSP messages for directory services are specified in Table 5. Figure 27 shows an example.. of usage of directory service messages.

**Table 5 – Messages for directory services**

| HDSP header |              |          | HDSP payload          |                       |                      |      |       |
|-------------|--------------|----------|-----------------------|-----------------------|----------------------|------|-------|
| TID         | DIR_QUE_REQ  | HDSPPLEN | PATHLEN<br>(4 bytes)  | PATH                  |                      |      |       |
| TID         | DIR_QUE_RES  | HDSPPLEN | ERRCODE<br>(4 bytes)  | DIRSNO<br>(4 bytes)   | FILESNO<br>(4 bytes) | DIRS | FILES |
| TID         | DIR_DEL_REQ  | HDSPPLEN | PATHLEN<br>(4 bytes)  | PATH                  |                      |      |       |
| TID         | DIR_DEL_RES  | HDSPPLEN | ERRCODE<br>(4 bytes)  |                       |                      |      |       |
| TID         | DIR_REN_REQ  | HDSPPLEN | PATHLEN1<br>(4 bytes) | PATHLEN2<br>(4 bytes) | PATH1                |      | PATH2 |
| TID         | DIR_REN_RES  | HDSPPLEN | ERRCODE<br>(4 bytes)  |                       |                      |      |       |
| TID         | DIR_MAKE_REQ | HDSPPLEN | PATHLEN<br>(4 bytes)  | PATH                  |                      |      |       |
| TID         | DIR_MAKE_RES | HDSPPLEN | ERRCODE<br>(4 bytes)  |                       |                      |      |       |

### 10.3.1 Query request message (DIR\_QUE\_REQ)

A DIR\_QUE\_REQ message is used for retrieving directory information of a remote device that also supports HDSP. HDSP payload of a DIR\_QUE\_REQ message shall be filled with PATHLEN and PATH fields, which are described below.

- PATHLEN: This field shall be four bytes long and shall be filled with byte size of PATH field. When the top directory is to be retrieved, this field shall be set to 0.
- PATH: This field shall be filled with a string of characters representing a full path to a directory of a remote CCP device which also supports HDSP. When the top directory is to be retrieved, this field shall be blanked.

### 10.3.2 Query response message (DIR\_QUE\_RES)

A DIR\_QUE\_RES message is a response of a DIR\_QUE\_REQ message. HDSP payload of a DIR\_QUE\_RES message shall be filled with ERRCODE, DIRSNO, FILESNO, DIRS and FILES fields, which are described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.
- DIRSNO: This field shall be four bytes long and shall be filled with the number of sub-directories in a retrieved directory.

- FILESNO: This field shall be four bytes long and shall be filled with the number of files in a retrieved directory.
- DIRS: This field shall be a field with a string of sub-directories' name in a retrieved directory. Each directory name of this field shall be separated with a horizontal tab (\t) character (0x09).
- FILES: This field shall be a field with a string of files' name in a retrieved directory. Each file name of this field shall be separated with a horizontal tab (\t) character (0x09).

### 10.3.3 Deletion request message (DIR\_DEL\_REQ)

A DIR\_DEL\_REQ message is used for deleting one directory of a remote device that also supports HDSP. HDSP payload of a DIR\_DEL\_REQ message shall be filled with PATHLEN and PATH fields, which are described below.

- PATHLEN: This field shall be four bytes long and shall be filled with byte size of PATH field.
- PATH: This field shall be filled with a string of characters representing a full path to a directory that is to be deleted.

### 10.3.4 Deletion response message (DIR\_DEL\_RES)

A DIR\_DEL\_RES message is a response of a DIR\_DEL\_REQ message. HDSP payload of a DIR\_DEL\_RES message shall be filled with ERRCODE field, which is described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.

### 10.3.5 Renaming request message (DIR\_REN\_REQ)

A DIR\_REN\_REQ message is used for renaming a directory of a remote device that also supports HDSP. HDSP payload of a DIR\_REN\_REQ message shall be filled with PATHLEN1, PATHLEN2, PATH1 and PATH2 fields, which are described below.

- PATHLEN1: This field shall be four bytes long and shall be filled with byte size of PATH1 field.
- PATHLEN2: This field shall be four bytes long and shall be filled with byte size of PATH2 field.
- PATH1: This field shall be filled with a string of characters representing a full path to a directory that is to be renamed.
- PATH2: This field shall be filled with a string of characters representing a full path to a renamed directory.

### 10.3.6 Renaming response message (DIR\_REN\_RES)

A DIR\_REN\_RES message is a response of a DIR\_REN\_REQ message. HDSP payload of a DIR\_REN\_RES message shall be filled with ERRCODE field, which is described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.

### 10.3.7 Making request message (DIR\_MAKE\_REQ)

A DIR\_MAKE\_REQ message is used for making a new directory of a remote device that also supports HDSP. HDSP payload of a DIR\_MAKE\_REQ message shall be filled with PATHLEN, and PATH fields, which are described below.

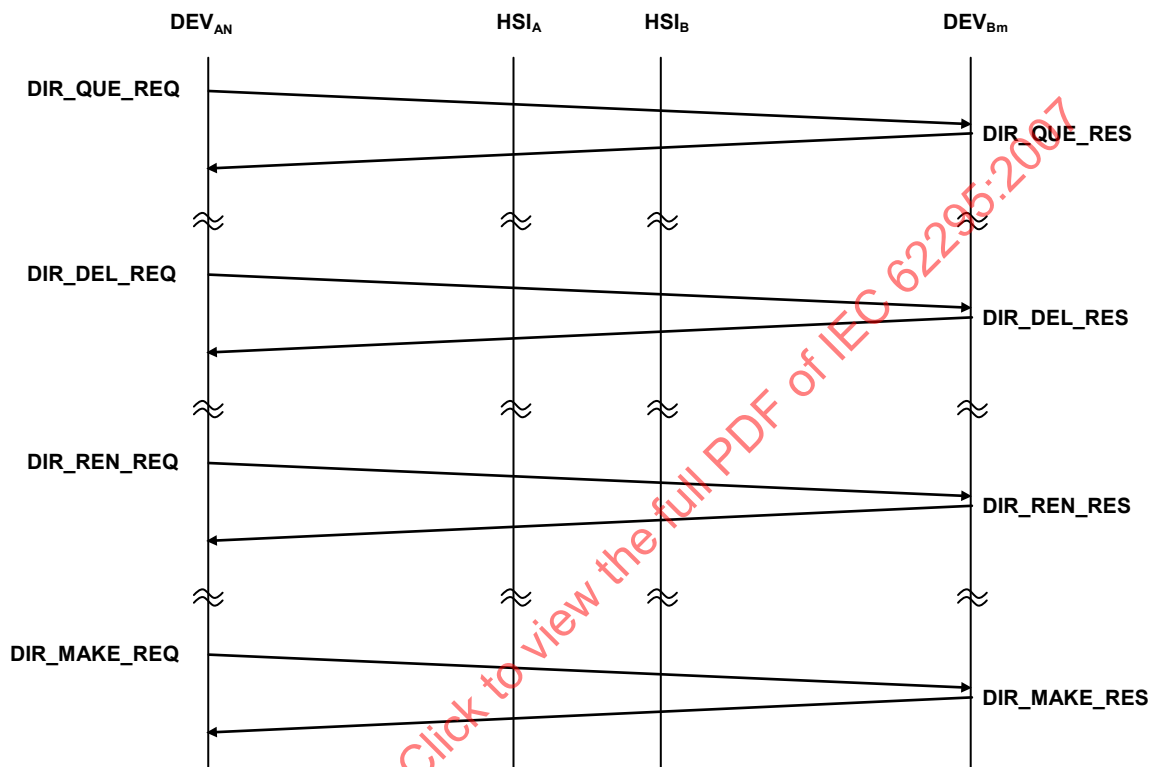
- PATHLEN: This field shall be four bytes long and shall be filled with byte size of PATH field.

- PATH: This field shall be filled with a string of characters representing a full path to a directory that is to be newly made.

### 10.3.8 Making response message (DIR\_MAKE\_RES)

A DIR\_MAKE\_RES message is a response of a DIR\_MAKE\_REQ message. HDSP payload of a DIR\_MAKE\_RES message shall be filled with ERRCODE field, which is described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.



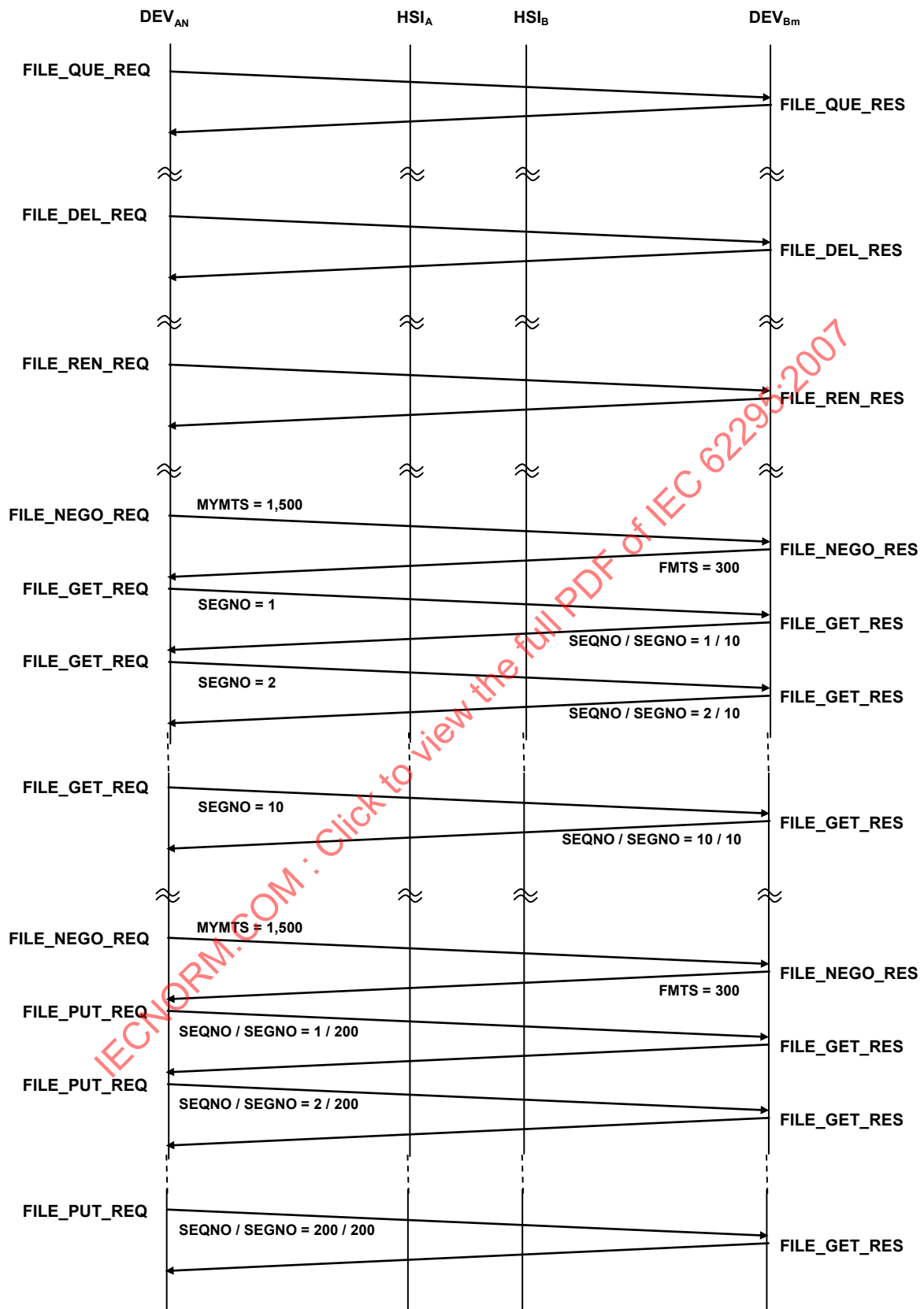
IEC 2097/07

Figure 27 – Example of usage of directory service messages

### 10.4 Messages for file services

HDSP messages for file services are specified in Table 6. Figure 28 shows an example of usage of file service messages.





IEC 2098/07

Figure 28 – Example of usage of file service messages

#### 10.4.1 Query request message (FILE\_QUE\_REQ)

A FILE\_QUE\_REQ message is used for retrieving information of a file stored in a remote device that also supports HDSP. HDSP payload of a FILE\_QUE\_REQ message shall be filled with PATHLEN and PATH fields, which are described below.

- PATHLEN: This field shall be four bytes long and shall be filled with byte size of PATH field.
- PATH: This field shall be filled with a string of characters representing a full path to a file that is to be retrieved.

#### 10.4.2 Query response message (FILE\_QUE\_RES)

A FILE\_QUE\_RES message is a response of a FILE\_QUE\_REQ message. HDSP payload of a FILE\_QUE\_RES message shall be filled with ERRCODE, FILESIZE, YEAR, MON, DAY, HOUR, MIN and SEC fields, which are described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.
- FILESIZE: This field shall be four bytes long and shall be filled with the size of a retrieved file in byte unit.
- YEAR, MON, DAY, HOUR, MIN and SEC: YEAR field shall be two bytes long. Other fields shall be one byte long. These fields shall represent the latest date and time when a retrieved file was modified.

#### 10.4.3 Deletion request message (FILE\_DEL\_REQ)

A FILE\_DEL\_REQ message is used for deleting a file stored in a remote device that also supports HDSP. HDSP payload of a FILE\_DEL\_REQ message shall be filled with PATHLEN and PATH fields, which are described below.

- PATHLEN: This field shall be four bytes long and shall be filled with byte size of PATH field.
- PATH: This field shall be filled with a string of characters representing a full path to a file that is to be deleted.

#### 10.4.4 Deletion response message (FILE\_DEL\_RES)

A FILE\_DEL\_RES message is a response of a FILE\_DEL\_REQ message. HDSP payload of a FILE\_DEL\_RES message shall be filled with ERRCODE field, which is described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.

#### 10.4.5 Renaming request message (FILE\_REN\_REQ)

A FILE\_REN\_REQ message is used for renaming a file stored in a remote device that also supports HDSP. HDSP payload of a FILE\_REN\_REQ message shall be filled with PATHLEN1, PATHLEN2, PATH1, and PATH2 fields, which are described below.

- PATHLEN1: This field shall be four bytes long and shall be filled with byte size of PATH1 field.
- PATHLEN2: This field shall be four bytes long and shall be filled with byte size of PATH2 field.
- PATH1: This field shall be filled with a string of characters representing a full path to a file that is to be renamed.
- PATH2: This field shall be filled with a string of characters representing a full path to a renamed file.

#### 10.4.6 Renaming response message (FILE\_REN\_RES)

A FILE\_REN\_RES message is a response of a FILE\_REN\_REQ message. HDSP payload of a FILE\_REN\_RES message shall be filled with ERRCODE field, which is described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.

#### 10.4.7 Negotiation request message (FILE\_NEGO\_REQ)

A FILE\_NEGO\_REQ message is used for negotiating maximum transfer size (MTS) with a remote device that also supports HDSP before a device initiates sending (or receiving) a file to (or from) another device. HDSP payload of a FILE\_NEGO\_REQ message shall be filled with MYMTS field, which is described below.

- MYMTS: This field shall be four bytes long and shall be filled with MTS supported and requested by a device that initiates file transfer. MYMTS shall be represented in byte unit.

#### 10.4.8 Negotiation response message (FILE\_NEGO\_RES)

A FILE\_NEGO\_RES message is a response of a FILE\_NEGO\_REQ message. HDSP payload of a FILE\_NEGO\_RES message shall be filled with ERRCODE and FMTS, which are described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.
- FMTS: This field shall be four bytes long and shall be filled with final negotiated MTS. In order to set this FMTS, a device receiving a FILE\_NEGO\_REQ message shall be able to compare MYMTS in the FILE\_NEGO\_REQ message with its own MTS and shall also be able to set FMTS to the lower MTS value between them. FMTS shall also be represented in byte unit.

#### 10.4.9 Getting request message (FILE\_GET\_REQ)

A FILE\_GET\_REQ message is used for copying a file from a remote device that also supports HDSP. Before sending this message, a device shall initiate negotiation for setting FMTS first. HDSP payload of a FILE\_GET\_REQ message shall be filled with FMTS, SEQNO, PATHLEN and PATH fields, which are described below.

- FMTS: This field shall be four bytes long and shall be set to the FMTS value in the FILE\_NEGO\_RES packet sent by a remote device.
- SEQNO: This field shall be four bytes long and shall indicate sequence number counted from 1 to SEGNO which will be set by a remote device.
- PATHLEN: This field shall be four bytes long and shall be filled with byte size of PATH field.
- PATH: This field shall be filled with a string of characters representing a full path to a file that is to be transferred from a remote device.

#### 10.4.10 Getting response message (FILE\_GET\_RES)

A FILE\_FET\_RES message is a response of a FILE\_GET\_REQ message. HDSP payload of a FILE\_GET\_RES message shall be filled with ERRCODE, SEGNO, SEQNO, DATALEN and DATA which are described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.
- SEGNO: This field shall be four bytes long and shall be filled with the total number of segments of a file that is to be transferred to the sender of the FILE\_GET\_REQ message.
- SEQNO: This field shall be four bytes long and shall be set to the SEQNO value of the FILE\_GET\_REQ message.

- DATALEN: This field shall be four bytes long and shall be set to the size of DATA field in byte unit. This field is usually set to the FMTS value of the FILE\_GET\_REQ message.
- DATA: This field shall be filled with one segment of a file to be transferred. The size of one segment is usually equal to FMTS or DATALEN value. A sender of this message shall be able to write one segment of a file by using SEQNO and FMTS values.

#### 10.4.11 Putting request message (FILE\_PUT\_REQ)

A FILE\_PUT\_REQ message is used for copying a local file to a remote device that also supports HDSP. Before sending this message, a device shall initiate negotiation for setting FMTS first. HDSP payload of a FILE\_PUT\_REQ message shall be filled with SEGNO, SEQNO, PATHLEN, DATALEN, PATH and DATA fields, which are described below.

- SEGNO: This field shall be four bytes long and shall be filled with the total number of segments of a file that is to be transferred to a remote device.
- SEQNO: This field shall be four bytes long and shall indicate the sequence number counted from 1.
- PATHLEN: This field shall be four bytes long and shall be filled with byte size of PATH field.
- DATALEN: This field shall be four bytes long and shall be set to the size of DATA field in byte unit. This field is usually set to the FMTS value in the FILE\_NEGO\_RES packet sent by a remote device.
- PATH: This field shall be filled with a string of characters representing a full path to a file stored in a remote device.
- DATA: This field shall be filled with one segment of a file to be transferred. The size of one segment is usually equal to FMTS or DATALEN value. A sender of this message shall be able to write one segment of a file by using SEQNO and FMTS values.

#### 10.4.12 Putting response message (FILE\_PUT\_RES)

A FILE\_PUT\_RES message is a response of a FILE\_PUT\_REQ message. HDSP payload of a FILE\_PUT\_RES message shall be filled with ERRCODE field, which is described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6.

### 10.5 Messages for messaging service

HDSP messages for messaging services are specified in Table. 7. Figure 29 shows an example of usage of messaging service messages.

**Table 7 – Messages for messaging services**

| HDSP header |                     |          | HDSP payload         |     |
|-------------|---------------------|----------|----------------------|-----|
| TID         | MSG_<br>PUT_<br>REQ | HDSPPLEN | MSGLEN<br>(4 bytes)  | MSG |
| TID         | MSG_<br>PUT_<br>RES | HDSPPLEN | ERRCODE<br>(4 bytes) |     |

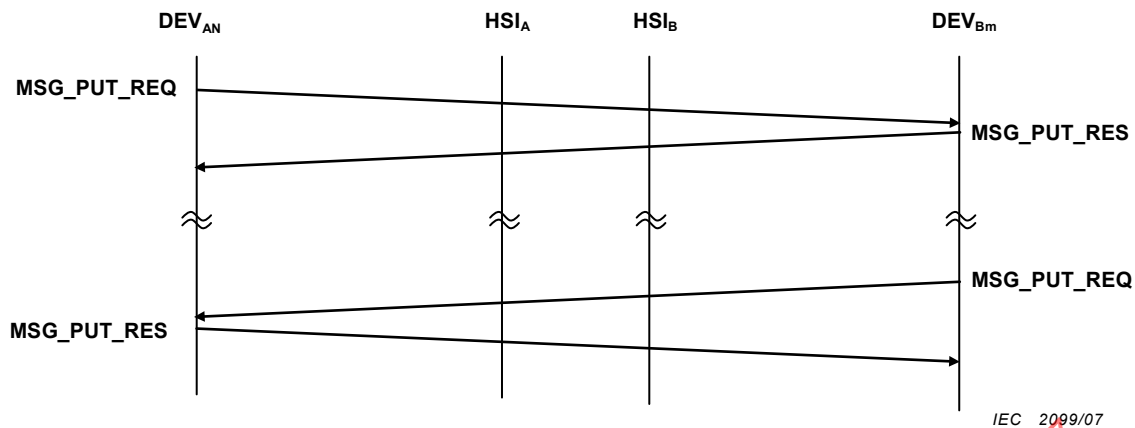


Figure 29 – Example of usage of Messaging service messages

#### 10.5.1 Sending request message (MSG\_PUT\_REQ)

A MSG\_PUT\_REQ message is used for sending a simple character-based message to a remote device that also supports HDSP. HDSP payload of a MSG\_PUT\_REQ message shall be filled with MSGLEN and MSG fields, which are described below.

- MSGLEN: This field shall be four bytes long and shall be filled with byte size of MSG field.
- MSG: This field shall be filled with a character-based message to be sent.

#### 10.5.2 Sending response message (MSG\_PUT\_RES)

A MSG\_PUT\_RES message is a response of a MSG\_PUT\_REQ message. HDSP payload of a MSG\_PUT\_RES message shall be filled with ERRCODE field, which is described below.

- ERRCODE: This field shall be four bytes long and shall be filled with an error code specified in 10.6. Error codes

#### 10.6 Error codes

Every HDSP response message shall return one of the error codes specified in Table. 8. These error codes are subject to change according to changes or modifications of HDSP messages.

**Table 8 – Error codes for HDSP**

| <b>ERRCODE<br/>(4 bytes)</b> | <b>Acronym</b>             | <b>Description</b>                                                                               |
|------------------------------|----------------------------|--------------------------------------------------------------------------------------------------|
| 0x03000000                   | HDSPERR_OK                 | A HDSP request message is processed properly without any error.                                  |
| 0x03010101                   | HDSPERR_DIR_NOT_FOUND      | Directory name of the HDSP request message can not be found.                                     |
| 0X03010102                   | HDSPERR_DIR_NOT_PERMITTED  | It is not permitted to access the directory because of an authority problem.                     |
| 0X03010103                   | HDSPERR_DIR_NOT_ACCESSED   | The directory is being accessed by another device or internal process.                           |
| 0X03010204                   | HDSPERR_DIR_NAME_INVALID   | One or more invalid character(s) is(are) used in the directory name of the HDSP request message. |
| 0X03010205                   | HDSPERR_DIR_NAME_LONG      | The directory name of the HDSP request message is too long.                                      |
| 0X03020101                   | HDSPERR_FILE_NOT_FOUND     | The file name of the HDSP request message can not be found.                                      |
| 0X03020102                   | HDSPERR_FILE_NOT_PERMITTED | It is not permitted to access the file because of an authority problem.                          |
| 0X03020103                   | HDSPERR_FILE_NOT_ACCESSED  | The file is being accessed by another device or internal process.                                |
| 0X03020204                   | HDSPERR_FILE_NAME_INVALID  | One or more invalid character(s) is(are) used in the file name of the HDSP request message.      |
| 0X03020205                   | HDSPERR_FILE_NAME_LONG     | The file name of the HDSP request message is too long.                                           |
| 0X03020301                   | HDSPERR_FILE_MTS_CHANGED   | The MTS value has been changed.                                                                  |
| 0X03020404                   | HDSPERR_FILE_SEGNO_INVALID | The SEGNO value in the HDSP request message is invalid.                                          |
| 0X03020504                   | HDSPERR_FILE_SEQNO_INVALID | The SEQNO value in the HDSP request message is invalid.                                          |
| 0X03020604                   | HDSPERR_FILE_DATA_INVALID  | The DATA filed in the HDSP request message is invalid.                                           |
| 0X03030706                   | HDSPERR_RESOURCE_NO_SPACE  | The remote device does not have enough space for creating or storing a file or directory.        |
| 0X03030807                   | HDSPERR_RESOURCE_TOO_BUSY  | The remote device is too busy to process the HDSP request packet.                                |
| OTHERS                       |                            | Reserved for future use.                                                                         |

## 11 Home multimedia service protocol (HMSP)

CCP provides the HMSP for offering multimedia services such as audio and video features under the home network environment.

HMSP is a packet-based signaling protocol, one of the fundamental protocols provided by the CCP. HMSP defines simple request/response packets and performs tasks such as management, real-time transmission and playing of multimedia resources within the home network.

All CCP devices that support multimedia services under the home network environment constructed with heterogeneous networks shall support the HMSP.

This document only specifies HMSP's functional requirements.

## 11.1 Functional requirements of HMSP

Devices and home servers that provide HMSP shall satisfy the following requirements.

### 11.1.1 Interoperability with CCP

In order to be compatible with CCP, HMSP shall satisfy the following requirements.

- HMSP shall be an application program using the CCP. Namely, commands or action results defined in relation with the HMSP shall be written in the CCP payload.
- The payload type field value in CCP header's type fields shall be 0x4.

### 11.1.2 Management of multimedia resource

HMSP shall provide functions that allow the user to access the various types of multimedia resources dispersed throughout numerous devices in the home network. The resources shall be readily available to the user regardless of time and place, as if they were stored in a single device.

In order to provide this kind of user-friendliness, HMSP shall satisfy the following requirements.

- HMSP shall provide functions for a device to register its multimedia resources information to the home server.
- HMSP shall provide functions for a particular device to request the home server for multimedia resources information for all devices within the home network.
- HMSP shall provide functions for the home server to respond to a particular device requesting information regarding multimedia resources of all devices within the home network.

NOTE Multimedia resources information includes the name, type, size, stored date and maximum transfer bandwidth of the resources.

### 11.1.3 Stream and play of multimedia resource

HMSP shall provide functions for streaming various types of multimedia resources stored in numerous devices in the home network as well as playing a multimedia stream being serviced by a particular device specified by the user.

In order to do so, HMSP shall satisfy the following requirements.

- HMSP shall provide functions for a particular device to request the home server for a streaming service of multimedia resources stored in another device. When doing so, the device making the streaming service request shall specify to the home server information such as the names of the multimedia resources, the maximum bandwidth it can handle and the qualities of video and audio desired by the user.
- HMSP shall provide functions for the home server to deliver the received streaming service request to the corresponding device with the multimedia resources.
- HMSP shall allow the device with the multimedia resources to provide the streaming service according to the request from the home server. The device receiving the request shall provide the streaming service only when it is feasible under the considerations of its processing power and the maximum bandwidth of its cluster network.
- If the service is deemed not feasible upon assessment of its processing capability and the maximum bandwidth of its cluster network, the device that has received the streaming service request from the home server shall be able to notify back to the home server with a status report indicating that the requested service cannot be provided. In turn, the home server shall be able to relay the received report to the device that had originally made the streaming service request.

- HMSP shall provide functions for a particular device to request information regarding the multimedia resources being serviced by the home server.
- HMSP shall provide functions for the home server to respond to a request made by a particular device for information regarding the multimedia resources being serviced by the home server.
- HMSP shall provide functions for a particular device to request multicasting of the multimedia resources being serviced.
- HMSP shall provide functions for the home server to accommodate a request made by a particular device for multicasting of the multimedia resources being serviced. However, multicasting shall be provided only when it is feasible upon the considerations of the home server's processing capability and the maximum bandwidth of the network that the requesting device is linked to.
- Upon receiving a request from a particular device to multicast the multimedia resources being serviced, if the home server determines that the multicasting is not feasible due to its processing capability and/or the maximum bandwidth of the network the requesting device is linked to, the home server shall be able to notify the requesting device with an infeasibility report.
- A device receiving multimedia resources shall have capabilities to play the resources.

IECNORM.COM : Click to view the full PDF of IEC 62295:2007

## Annex A (informative)

### FSM of FS-CCPDEV supporting HNMP

#### A.1 Finite state machine (FSM) for HMSP

Figure A.1 displays the finite state machine (FSM) of an FS-CCPDEV device that supports the HNMP.

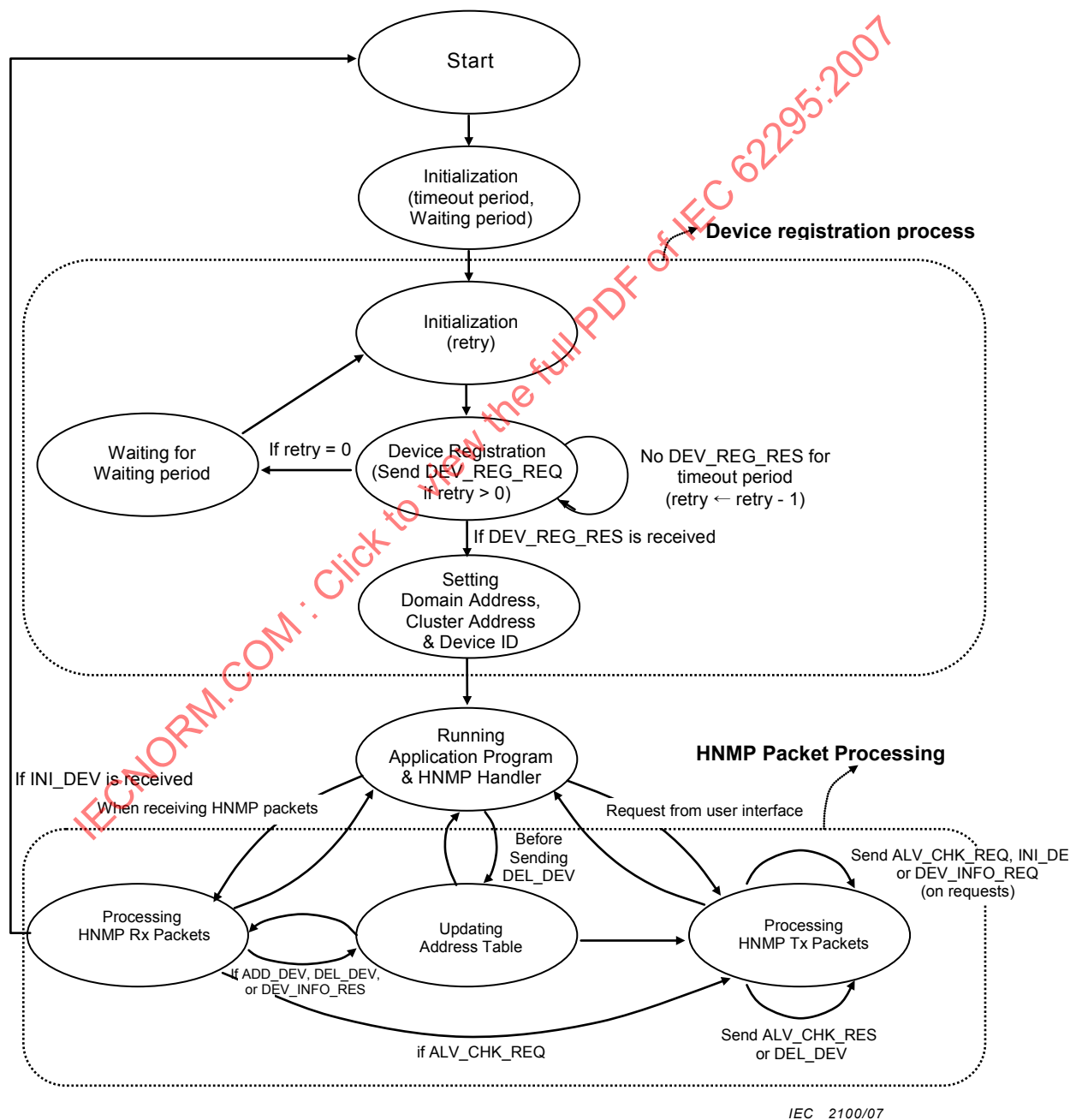


Figure A.1 – FSM of FS-CCPDEV for supporting HNMP

## A.2 Start

After the FS-CCPDEV is powered on, it starts the initialization process. If the FS-CCPDEV receives an INI\_DEV packet during normal operation, it shall be able to re-initialize itself. A device that has completed the device registration process does not have to register itself again after initialization.

## A.3 Initialization

The retry, timeout and waiting variables are initialized. Since these variables are subject to the processing power of a device and the network speed, device manufacturers or developers are recommended to initialize these variables to appropriate constants. If the FS-CCPDEV does not receive a DEV\_REG\_RES packet within the timeout period, it attempts to send DEV\_REG\_REQ packets  $N$  times, where  $N$  is equal to the retry variables.

## A.4 Device registration

Upon completion of initialization, the FS-CCPDEV sends a DEV\_REG\_REQ packet to the HSI of the cluster network it belongs to, in order to request the domain address, cluster address and its own device ID. When doing so, the FS-CCPDEV writes its name in the DEV\_REG\_REQ packet payload prior to transmission.

## A.5 Waiting

After sending a DEV\_REG\_REQ packet, the FS-CCPDEV waits for a DEV\_REG\_RES packet from the HSI. If there is no response within timeout period, the FS-CCPDEV attempts retries. If there is still no response after the retries, the home server is regarded as not being ready, and the device registration process is repeated after a waiting period.

## A.6 Setting domain address and device ID

After receiving the DEV\_REG\_RES packet, the FS-CCPDEV configures domain address, cluster address, device ID and HSI's physical address (or network address).

## A.7 Running application program and HNMP handler

Upon successful completion of the device registration process, the FS-CCPDEV simultaneously executes handlers for the application program and HNMP packet processing.

## A.8 Processing Rx HNMP packets

If the HNMP handler receives an HNMP packet during application program execution, the HNMP handler processes the received HNMP packet. If the packet is an INI\_DEV, device initialization is performed. If the received packet is an ADD\_DEV, DEL\_DEV or DEV\_INFO\_RES, the HNMP handler updates its address table. If the packet is an ALV\_CHK\_REQ, the handler returns an ALV\_CHK\_RES packet.

## A.9 Processing Tx HNMP packets

The user shall be able to send an ALV\_CHK\_REQ or INI\_DEV to another device within the home network through the user interface. In addition, the user shall be able to send a DEV\_INFO\_REQ to the HSI to request for address information of the home network devices. If an ALV\_CHK\_REQ is sent according to the user request and an ALV\_CHK\_RES packet is

not returned, the process is retried  $N$  times, where  $N$  is equal to the retry variable. If an ALV\_CHK\_RES packet still fails to be returned, a DEL\_DEV shall be sent as a broadcast. Moreover, when an ALV\_CHK\_REQ is received, an ALV\_CHK\_RES packet shall be returned as a response within a timeout period.

#### **A.10 Updating address table**

When the FS-CCPDEV receives an ADD\_DEV, DEL\_DEV or DEV\_INFO\_RES packet, it shall be able to update the address table it manages.

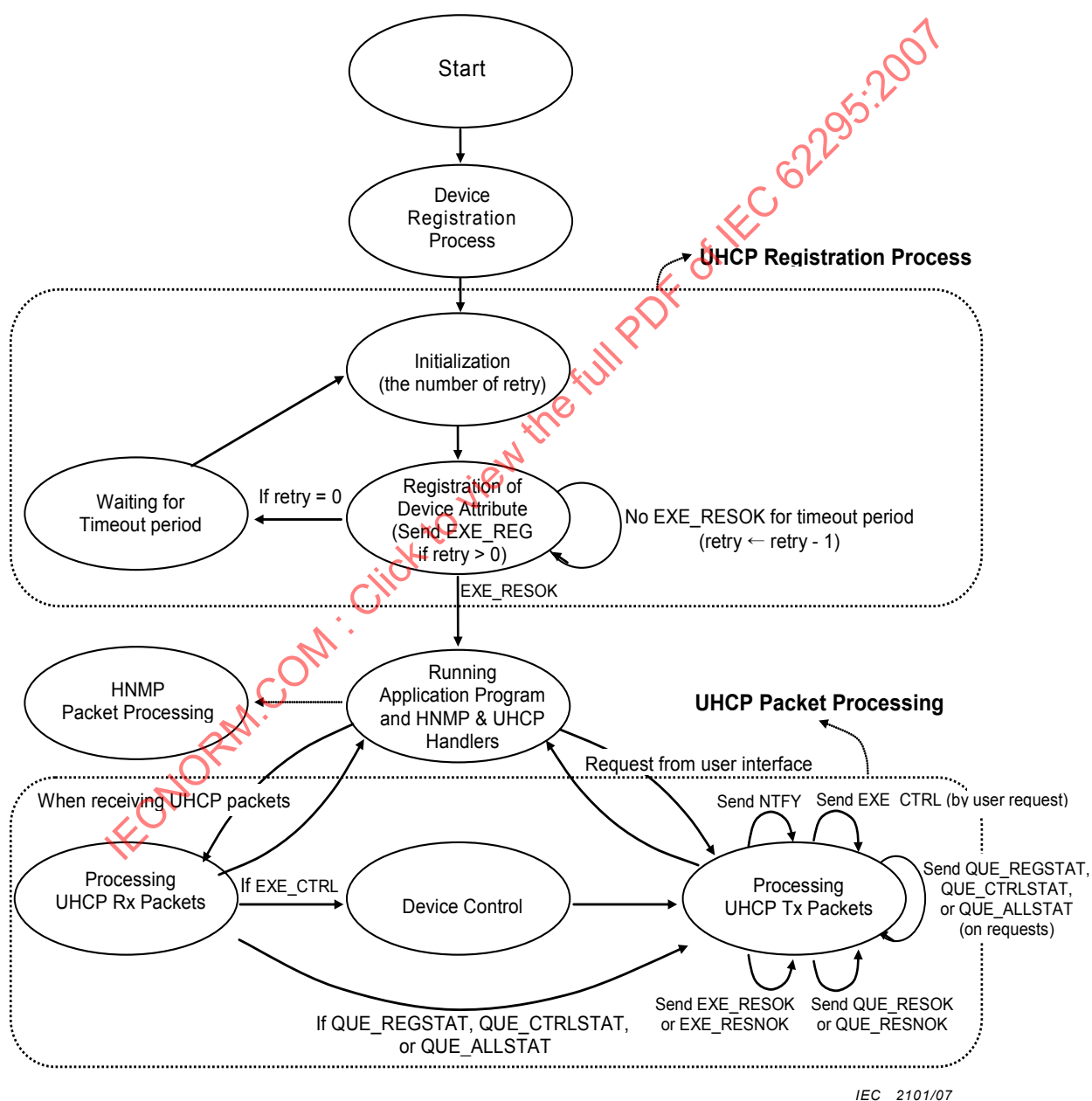
IECNORM.COM : Click to view the full PDF of IEC 62295:2007

## Annex B (informative)

### FSM of FS-CCPDEV for supporting UHCP

#### B.1 Finite state machine (FSM) for UHCP

Figure B.1 displays the finite state machine (FSM) of an FS-CCPDEV with UHCP functions implemented.



IEC 2101/07

**Figure B.1 – FSM of FS-CCPDEV for supporting UHCP**

#### B.2 Start

Refer to Annex A.1.

### **B.3 Device registration process**

Refer to Annex A from A.2 to A.6.

### **B.4 Initialization**

The retry variable is initialized. If the FS-CCPDEV does not receive an EXE\_RESOK packet within the timeout period, it attempts retries  $N$  times, where  $N$  is equal to the retry variable.

### **B.5 Registration of device attribute**

Upon completion of initialization, the FS-CCPDEV sends an EXE\_REG packet to the HSI of the home server, in order to register its device attributes and control or monitor items in the home server.

### **B.6 Waiting**

If the FS-CCPDEV does not receive an EXE\_RESOK response packet within the timeout period, the FS-CCPDEV attempts retries. If there is still no response after retries, the home server is regarded as not being ready, and the device attribute registration process is repeated after a waiting period.

### **B.7 Running application program and HNMP and UHCP handlers**

Upon successful completion of the device attribute registration process, the FS-CCPDEV simultaneously executes handlers for the application program, HNMP packet processing and UHCP message processing.

### **B.8 Processing UHCP Rx packets**

If the UHCP handler receives an UHCP packet during application program execution, the UHCP handler processes the received UHCP packet. If the packet is an EXE\_CTRL, the device itself is controlled according to the control command. If the received packet is a query request command such as QUE\_REGSTAT, QUE\_CTRLSTAT or QUE\_ALLSTAT, the HNMP handler writes its current registration status and control status in the QUE\_RESOK payload and returns the packet. If the requested query is not feasible, the handler returns a QUE\_RESNOK packet.

### **B.9 Processing UHCP Tx packets**

The user shall be able to send control request commands such as EXE\_CTRL and query request commands such as QUE\_REGSTAT, QUE\_CTRLSTAT or QUE\_ALLSTAT to another device within the home network through the user interface. On the other hand, if the FS-CCPDEV receives a control request command, it shall be able to return an EXE\_RESOK or EXE\_RESNOK packet. Furthermore, the FS-CCPDEV shall be able to notify its abnormal status with the NTFY packet.

### **B.10 Device control**

When the FS-CCPDEV's UHCP handler receives an EXE\_CTRL packet, it shall be able to control itself according to the content of the EXE\_CTRL packet payload.

IECNORM.COM : Click to view the full PDF of IEC 62295:2007

## SOMMAIRE

|                                                                                                                               |    |
|-------------------------------------------------------------------------------------------------------------------------------|----|
| AVANT-PROPOS.....                                                                                                             | 71 |
| INTRODUCTION.....                                                                                                             | 73 |
| 1 Domaine d'application et objet.....                                                                                         | 74 |
| 2 Références normatives.....                                                                                                  | 75 |
| 3 Termes, définitions, abréviations et conventions.....                                                                       | 75 |
| 3.1 Termes et définitions.....                                                                                                | 75 |
| 3.2 Abréviations.....                                                                                                         | 78 |
| 3.3 Conventions.....                                                                                                          | 80 |
| 4 Exigences.....                                                                                                              | 80 |
| 4.1 Exigences concernant l'interface de serveur domestique.....                                                               | 80 |
| 4.1.1 Exigences de base pour la fourniture de données.....                                                                    | 80 |
| 4.1.2 Exigences fonctionnelles concernant le HNMP.....                                                                        | 80 |
| 4.1.3 Exigences concernant l'interface de serveur domestique pour<br>monodiffusion, multidiffusion et diffusion générale..... | 81 |
| 4.2 Exigences concernant le dispositif de CCP.....                                                                            | 81 |
| 4.2.1 Exigences concernant l'enregistrement de dispositif.....                                                                | 81 |
| 4.2.2 Exigences concernant la classification des dispositifs de CCP.....                                                      | 82 |
| 5 Couche de protocole de communication commun (CCP).....                                                                      | 83 |
| 5.1 Couche CCP.....                                                                                                           | 83 |
| 5.2 Fourniture de données sur des réseaux hétérogènes.....                                                                    | 85 |
| 6 Adressage de CCP.....                                                                                                       | 86 |
| 6.1 Généralités.....                                                                                                          | 86 |
| 6.2 Structure d'adressage facilitant la commutation du trafic pour le serveur<br>domestique Version 0.....                    | 86 |
| 6.2.1 Domain address (Adresse de domaine).....                                                                                | 87 |
| 6.2.2 Cluster address (Adresse de grappe).....                                                                                | 87 |
| 6.2.3 Champ Device ID (Identificateur de dispositif).....                                                                     | 87 |
| 7 Format et champs des paquets CCP.....                                                                                       | 88 |
| 7.1 Généralités.....                                                                                                          | 88 |
| 7.2 Format des paquets CCP.....                                                                                               | 88 |
| 7.2.1 CCP identification (Identification de CCP) (CCPID).....                                                                 | 88 |
| 7.2.2 CCP header version (Version de l'en-tête CCP) (CCPHDRVER).....                                                          | 89 |
| 7.2.3 CCP address version (Version de l'adresse CCP) (CCPADDRVER).....                                                        | 89 |
| 7.2.4 Destination address (Adresse destination) (DESTADDR).....                                                               | 89 |
| 7.2.5 Source address (Adresse source) (SRCADDR).....                                                                          | 89 |
| 7.2.6 Champ Type (TYPE).....                                                                                                  | 89 |
| 7.2.7 Champ Reserved (Réservé) (RSV).....                                                                                     | 91 |
| 7.2.8 Champ CCP payload length (Longueur de charge utile CCP)<br>(CCPPLEN).....                                               | 91 |
| 7.2.9 Champ CCP payload (Charge utile CCP).....                                                                               | 92 |
| 8 Protocole de gestion de réseau domestique (HNMP).....                                                                       | 92 |
| 8.1 Généralités.....                                                                                                          | 92 |
| 8.2 Format des paquets HNMP.....                                                                                              | 92 |
| 8.2.1 Transaction ID (Identificateur de transaction) (TID).....                                                               | 92 |
| 8.2.2 HNMP command (Commande HNMP) (HNMPCMD).....                                                                             | 93 |

|       |                                                                                                                                             |     |
|-------|---------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 8.2.3 | Champ Reserved (Réservé) (RSV) .....                                                                                                        | 93  |
| 8.2.4 | Champ HNMP payload length (Longueur de charge utile HNMP)<br>(HNMPPLEN) .....                                                               | 93  |
| 8.2.5 | HNMP payload (Charge utile HNMP) .....                                                                                                      | 93  |
| 8.3   | Enregistrement d'un serveur domestique .....                                                                                                | 93  |
| 8.4   | Enregistrement d'un dispositif .....                                                                                                        | 93  |
| 8.4.1 | Paquet Device registration request (Requête d'enregistrement de<br>dispositif) (DEV_REG_REQ) .....                                          | 94  |
| 8.4.2 | Paquet Device registration response (Réponse à une requête<br>d'enregistrement de dispositif) (DEV_REG_RES) .....                           | 95  |
| 8.5   | Gestion de dispositif .....                                                                                                                 | 96  |
| 8.5.1 | Paquet Add Device (Ajout de dispositif) (ADD_DEV) .....                                                                                     | 98  |
| 8.5.2 | Paquet Delete device (Suppression de dispositif) (DEL_DEV) .....                                                                            | 98  |
| 8.5.3 | Paquet Initialize Device (Initialiser dispositif) (INI_DEV) .....                                                                           | 99  |
| 8.5.4 | Paquet Alive-Check Request (Requête de vérification<br>d'activité) (ALV_CHK_REQ) .....                                                      | 99  |
| 8.5.5 | Paquet Alive-Check Response (Réponse à une requête de<br>vérification d'activité) (ALV_CHK_RES) .....                                       | 99  |
| 8.6   | Informations d'adresse et de nom des dispositifs .....                                                                                      | 99  |
| 8.6.1 | Paquet Device address and name information request (Requête<br>d'informations d'adresse et de nom de dispositif) (DEV_INFO_REQ) .....       | 101 |
| 8.6.2 | Paquet Device address and name information response (Réponse<br>aux informations d'adresse et de nom de<br>dispositif) (DEV_INFO_RES) ..... | 101 |
| 8.7   | Autres fonctions de gestion .....                                                                                                           | 102 |
| 9     | Protocole de contrôle domestique universel (UHCP) .....                                                                                     | 102 |
| 9.1   | Format des paquets UHCP .....                                                                                                               | 102 |
| 9.1.1 | Transaction ID (identificateur de transaction) (TID) .....                                                                                  | 102 |
| 9.1.2 | Message type (type de message) (MT) et action type (type<br>d'action) (AT) .....                                                            | 103 |
| 9.1.3 | Champ reserved (réservé) (RSV) .....                                                                                                        | 103 |
| 9.1.4 | Champ UHCP payload length (longueur de charge utile UHCP)<br>(UHCPPLEN) .....                                                               | 103 |
| 9.1.5 | UHCP payload (charge utile UHCP) .....                                                                                                      | 104 |
| 9.2   | Messages execution (exécution) (EXE) .....                                                                                                  | 104 |
| 9.2.1 | Exécution de l'enregistrement (EXE_REG) .....                                                                                               | 104 |
| 9.2.2 | Exécution du contrôle (EXE_CTRL) .....                                                                                                      | 105 |
| 9.2.3 | Response OK (Réponse correcte) (EXE_RESOK) .....                                                                                            | 105 |
| 9.2.4 | Response NOK (Réponse incorrecte) (EXE_RESNOK) .....                                                                                        | 105 |
| 9.3   | Messages query (demande) (QUE) .....                                                                                                        | 105 |
| 9.3.1 | Query of registration status (demande d'état<br>d'enregistrement) (QUE_REGSTAT) .....                                                       | 106 |
| 9.3.2 | Query of control status (demande d'état de<br>contrôle) (QUE_CTRLSTAT) .....                                                                | 106 |
| 9.3.3 | Query of all status (demande de tous les états) (QUE_ALLSTAT) .....                                                                         | 107 |
| 9.3.4 | Response OK (Réponse correcte) (QUE_RESOK) .....                                                                                            | 107 |
| 9.3.5 | Response NOK (Réponse incorrecte) (QUE_RESNOK) .....                                                                                        | 107 |
| 9.4   | Messages notification (notification) (NTFY) .....                                                                                           | 107 |
| 9.5   | Syntaxe de la charge utile UHCP .....                                                                                                       | 108 |
| 9.5.1 | Syntaxe de base pour la charge utile UHCP .....                                                                                             | 108 |
| 9.5.2 | Syntaxe de l'enregistrement UHCP .....                                                                                                      | 108 |

|         |                                                                                             |     |
|---------|---------------------------------------------------------------------------------------------|-----|
| 9.5.3   | Syntaxe pour le contrôle d'un dispositif.....                                               | 109 |
| 9.5.4   | Syntaxe pour la demande d'état de contrôle et de surveillance .....                         | 110 |
| 9.5.5   | Syntaxe pour la notification .....                                                          | 111 |
| 10      | Protocole de service de données domestique (HDSP), .....                                    | 112 |
| 10.1    | Exigences fonctionnelles du HDSP.....                                                       | 112 |
| 10.1.1  | Interopérabilité avec CCP.....                                                              | 112 |
| 10.1.2  | Services de fichiers et de répertoires.....                                                 | 112 |
| 10.1.3  | Service de messagerie .....                                                                 | 113 |
| 10.2    | Format des paquets HDSP .....                                                               | 113 |
| 10.2.1  | Transaction ID (Identificateur de transaction) (TID) .....                                  | 113 |
| 10.2.2  | HDSP command (Commande HDSP).....                                                           | 113 |
| 10.2.3  | Champ HDSP payload length (Longueur de charge utile HDSP)<br>(HDSPPLEN).....                | 114 |
| 10.2.4  | Charge utile HDSP .....                                                                     | 114 |
| 10.3    | Messages pour les services de répertoire.....                                               | 114 |
| 10.3.1  | Message Query request (Requête de demande) (DIR_QUE_REQ) .....                              | 115 |
| 10.3.2  | Message Query response (Réponse à une requête de demande)<br>(DIR_QUE_RES) .....            | 115 |
| 10.3.3  | Message Deletion request (Requête de suppression)<br>(DIR_DEL_REQ) .....                    | 116 |
| 10.3.4  | Message Deletion response (Réponse à une requête de suppression)<br>(DIR_DEL_RES).....      | 116 |
| 10.3.5  | Message Renaming request (Requête de renommage)<br>(DIR_REN_REQ).....                       | 116 |
| 10.3.6  | Message Renaming response (Réponse à une requête de<br>renommage) (DIR_REN_RES).....        | 116 |
| 10.3.7  | Message Making request (Requête de création) (DIR_MAKE_REQ) .....                           | 116 |
| 10.3.8  | Message Making response (Réponse à une requête de création)<br>(DIR_MAKE_RES).....          | 117 |
| 10.4    | Messages pour les services de fichier .....                                                 | 117 |
| 10.4.1  | Message Query request (Requête de demande) (FILE_QUE_REQ).....                              | 120 |
| 10.4.2  | Message Query response (Réponse à une requête de demande)<br>(FILE_QUE_RES) .....           | 120 |
| 10.4.3  | Message Deletion request (Requête de suppression)<br>(FILE_DEL_REQ).....                    | 120 |
| 10.4.4  | Message Deletion response (Réponse à une requête de<br>suppression) (FILE_DEL_RES) .....    | 120 |
| 10.4.5  | Message Renaming request (Requête de renommage)<br>(FILE_REN_REQ) .....                     | 120 |
| 10.4.6  | Message Renaming response (Réponse à une requête de<br>renommage) (FILE_REN_RES) .....      | 121 |
| 10.4.7  | Message Negotiation request (Requête de négociation)<br>(FILE_NEGO_REQ) .....               | 121 |
| 10.4.8  | Message Negotiation response (Réponse à une requête de<br>négociation) (FILE_NEGO_RES)..... | 121 |
| 10.4.9  | Message Getting request (Requête d'obtention) (FILE_GET_REQ) .....                          | 121 |
| 10.4.10 | Message Getting response (Réponse à une requête d'obtention)<br>(FILE_GET_RES).....         | 122 |
| 10.4.11 | Message Putting request (Requête de placement)<br>(FILE_PUT_REQ).....                       | 122 |
| 10.4.12 | Message Putting response (Réponse à une requête de placement)<br>(FILE_PUT_RES).....        | 122 |

|                                                                                                                                                                                   |                                                                                 |     |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|-----|
| 10.5                                                                                                                                                                              | Messages pour le service de messagerie .....                                    | 123 |
| 10.5.1                                                                                                                                                                            | Message Sending request (Requête d'envoi) (MSG_PUT_REQ) .....                   | 123 |
| 10.5.2                                                                                                                                                                            | Message Sending response (Réponse à une requête d'envoi)<br>(MSG_PUT_RES) ..... | 123 |
| 10.6                                                                                                                                                                              | Codes d'erreur .....                                                            | 123 |
| 11                                                                                                                                                                                | Protocole de service multimédia domestique (HMSP) .....                         | 124 |
| 11.1                                                                                                                                                                              | Exigences fonctionnelles du HMSP .....                                          | 125 |
| 11.1.1                                                                                                                                                                            | Interopérabilité avec CCP .....                                                 | 125 |
| 11.1.2                                                                                                                                                                            | Gestion d'une ressource multimédia .....                                        | 125 |
| 11.1.3                                                                                                                                                                            | Diffusion en flux et lecture d'une ressource multimédia .....                   | 125 |
| Annexe A (informative) Machine à états finis (FSM) d'un dispositif de CCP (FS-CCPDEV) prenant en charge HNMP .....                                                                |                                                                                 | 127 |
| Annexe B (informative) Machine à états finis (FSM) d'un dispositif de CCP (FS-CCPDEV) prenant en charge UHCP .....                                                                |                                                                                 | 130 |
| Figure 1 – Structures des couches de communication des technologies de réseau .....                                                                                               |                                                                                 | 75  |
| Figure 2 – Interface de serveur .....                                                                                                                                             |                                                                                 | 76  |
| Figure 3 – Réseau en grappe et réseau de domaine .....                                                                                                                            |                                                                                 | 77  |
| Figure 4 – Classification des dispositifs de CCP .....                                                                                                                            |                                                                                 | 82  |
| Figure 5 – Définitions du programme d'application, Interface de programmation (API) CCP, interface des couches de protocole inférieures et couches de protocole inférieures ..... |                                                                                 | 84  |
| Figure 6 – Emplacement de la couche CCP .....                                                                                                                                     |                                                                                 | 84  |
| Figure 7 – Exemple de transmission de données sur des réseaux hétérogènes utilisant la couche CCP .....                                                                           |                                                                                 | 86  |
| Figure 8 – Format d'adresse CCP d'une adresse CCP version 0 .....                                                                                                                 |                                                                                 | 87  |
| Figure 9 – Format des en-têtes de paquet CCP, version 0. ....                                                                                                                     |                                                                                 | 88  |
| Figure 10 – Champs Type .....                                                                                                                                                     |                                                                                 | 89  |
| Figure 11 – Format des paquets HNMP .....                                                                                                                                         |                                                                                 | 92  |
| Figure 12 – Paquets DEV_REG_REQ et DEV_REG_RES .....                                                                                                                              |                                                                                 | 95  |
| Figure 13 – Exemple de séquence de commandes HNMP pour enregistrement de dispositif .....                                                                                         |                                                                                 | 96  |
| Figure 14 – Paquets ADD_DEV, DEL_DEV et INI_DEV .....                                                                                                                             |                                                                                 | 97  |
| Figure 15 – Paquets ALV_CHK_REQ et ALV_CHK_RES .....                                                                                                                              |                                                                                 | 97  |
| Figure 16 – Exemple de séquence de commandes HNMP pour la gestion de dispositif .....                                                                                             |                                                                                 | 98  |
| Figure 17 – Paquets DEV_INFO_REQ et DEV_INFO_RES .....                                                                                                                            |                                                                                 | 100 |
| Figure 18 – Exemple de séquence de commandes HNMP pour récupérer les informations d'adresse et de nom de dispositif .....                                                         |                                                                                 | 101 |
| Figure 19 – Format des paquets UHCP .....                                                                                                                                         |                                                                                 | 102 |
| Figure 20 – Champs message type (type de message) et action type (type d'action) d'un paquet UHCP .....                                                                           |                                                                                 | 103 |
| Figure 21 – Exemple de processus d'enregistrement .....                                                                                                                           |                                                                                 | 104 |
| Figure 22 – Exemple de message EXE_CTRL .....                                                                                                                                     |                                                                                 | 105 |
| Figure 23 – Exemple de message QUE_REGSTAT .....                                                                                                                                  |                                                                                 | 106 |
| Figure 24 – Exemple de message QUE_CTRLSTAT .....                                                                                                                                 |                                                                                 | 106 |
| Figure 25 – Exemple de message QUE_ALLSTAT .....                                                                                                                                  |                                                                                 | 107 |

|                                                                               |     |
|-------------------------------------------------------------------------------|-----|
| Figure 26 – Format des paquets HDSP .....                                     | 113 |
| Figure 27 – Exemple d'utilisation des messages de service de répertoire ..... | 117 |
| Figure 28 – Exemple d'utilisation des messages de service de fichier .....    | 119 |
| Figure 29 – Exemple d'utilisation des messages de service de messagerie. .... | 123 |
| Figure A.1 – FSM de FS-CCPDEV prenant en charge HNMP .....                    | 127 |
| Figure B.1 – FSM de FS-CCPDEV prenant en charge UHCP .....                    | 130 |
|                                                                               |     |
| Tableau 1 – Champ Cast type (Type de diffusion) .....                         | 90  |
| Tableau 2 – Champ Traffic type (Type de trafic) .....                         | 91  |
| Tableau 3 – Champ Payload type (Type de charge utile) .....                   | 91  |
| Tableau 4 – Commandes HDSP .....                                              | 114 |
| Tableau 5 – Messages pour les services de répertoire .....                    | 115 |
| Tableau 6 – Messages pour les services de fichier .....                       | 118 |
| Tableau 7 – Messages pour les services de messagerie .....                    | 123 |
| Tableau 8 – Codes d'erreur pour HDSP .....                                    | 124 |

IECNORM.COM : Click to view the full PDF of IEC 62295:2007

## COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

**SYSTÈMES MULTIMÉDIA –  
PROTOCOLE DE COMMUNICATION COMMUN  
RELATIF À LA CONNECTIVITÉ ENTRE RÉSEAUX HÉTÉROGÈNES**

## AVANT-PROPOS

- 1) La Commission Electrotechnique Internationale (CEI) 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 les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI 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. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 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.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. 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.

La Norme internationale CEI 62295 a été établie par le domaine technique 8: Systèmes de serveurs domestiques multimédia, du comité d'études 100 de la CEI: Systèmes et appareils audio, vidéo et multimédia.

La présente version bilingue (2012-08) correspond à la version anglaise monolingue publiée en 2007-11.

Le texte anglais de cette norme est issu des documents 100/1200/CDV et 100/1283/RVC.

Le rapport de vote 100/1283/RVC donne toute information sur le vote ayant abouti à l'approbation de cette norme.

La version française n'a pas été soumise au vote.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de maintenance indiquée sur le site web de la CEI sous "<http://webstore.iec.ch>" dans les données relatives à la publication recherchée. A cette date, la publication sera

- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

**IMPORTANT – Le logo "colour inside" qui se trouve sur la page de couverture de cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.**

IECNORM.COM : Click to view the full PDF of IEC 62295:2007

## INTRODUCTION

Un grand nombre de technologies de réseaux domestiques câblés et sans fil de types divers ont été élaborées et sont actuellement utilisées. Toutefois, puisque ces technologies ont été élaborées pour des fonctions spécifiques telles que des services de contrôle, A/V (Audiovisuel) et de données, l'interopérabilité n'est pas garantie entre les produits utilisant ces technologies. Ainsi, les utilisateurs souhaitant mettre en œuvre un environnement de réseau domestique achètent des dispositifs reposant sur une technologie d'interopérabilité unique ou installent à leur domicile des réseaux indépendants et incompatibles.

Pour résoudre ces problèmes, l'industrie des réseaux domestiques et les fournisseurs de services ont pris en compte et ont mis au point un certain nombre de technologies spécifiques afin d'assurer l'interopérabilité entre les technologies de réseaux domestiques. Toutefois, la plupart de ces technologies sont locales et proposent une interopérabilité entre un nombre limité de dispositifs, ce qui fait apparaître de nouveaux problèmes dus à la complexité et à la diversité des technologies de différents fournisseurs.

Pour tenir compte de ces technologies complexes et diverses, un besoin existe pour mettre au point une nouvelle technologie de convergence intégrant non seulement les technologies actuelles mais également celles dont on s'attend à ce qu'elles apparaissent dans l'avenir.

Les besoins de cette nouvelle technologie de convergence sont les suivants:

- assurer l'interopérabilité et l'interconnectivité entre des réseaux hétérogènes par l'intermédiaire d'une couche de convergence spécifique;
- assurer une capacité d'extension pour des applications, non seulement dans les technologies de réseaux actuelles, mais également dans les nouvelles technologies qui seront mises au point dans l'avenir;
- application à des petits dispositifs avec capacités de traitement réduites, en fournissant des protocoles tels qu'une signalisation simple dans la couche de convergence;
- disponibilité à un coût réduit et mise en œuvre simple sur un dispositif;
- aptitude à fournir divers services (ou applications) de réseaux domestiques.

## SYSTÈMES MULTIMÉDIA – PROTOCOLE DE COMMUNICATION COMMUN RELATIF À LA CONNECTIVITÉ ENTRE RÉSEAUX HÉTÉROGÈNES

### 1 Domaine d'application et objet

La présente Norme internationale spécifie la couche de protocole de communication commun (CCP) capable d'assurer l'interopérabilité et l'interconnectivité entre des technologies de réseau hétérogènes, ainsi que la configuration de transmission de base des données entre des dispositifs reliés à des réseaux hétérogènes par l'intermédiaire de la couche CCP. La présente norme spécifie également la structure des paquets dans la couche CCP et le système d'adressage commun pouvant être compris entre les dispositifs hétérogènes. De plus, il existe des spécifications concernant des protocoles capables de fournir diverses aux applications de réseau domestique par l'intermédiaire de la couche CCP, tels que le Protocole de gestion de réseau domestique (HNMP), le Protocole de contrôle domestique universel (UHCP), le Protocole de services multimédias domestiques (HMSP) et le Protocole de services de données domestiques (HDSP).

NOTE HNMP est le protocole de gestion de réseau domestique global qui détecte ou enregistre des dispositifs. UHCP contrôle et surveille des dispositifs à des emplacements distants. HMSP est le protocole A/V pour les services domestiques de divertissement. HDSP traite les services de données et de messagerie.

La présente norme est destinée à être appliquée à des systèmes dotés de capacités de réseau tels que ceux qui constituent des réseaux domestiques, par exemple des appareils électroniques, des composants A/V, des dispositifs de contrôle, des terminaux de réseau et des serveurs domestiques. La présente norme s'applique également à un réseau domestique constitué d'un unique serveur domestique.

La présente norme internationale donne

- une définition de la couche CCP,
- une configuration de transmission de données dans la couche CCP,
- une structure de paquet CCP,
- une configuration d'adressage de CCP,
- un protocole de gestion de réseau domestique (HNMP),
- un protocole de contrôle domestique universel (UHCP),
- un protocole de services de données domestiques (HDSP),
- des exigences concernant un protocole de services multimédias domestiques (HMSP).

Un réseau domestique assure l'interopérabilité et l'interconnectivité, quel que soit le fabricant des appareils ou le type de réseau, de telle sorte que l'utilisateur est capable de recevoir à tout moment les services désirés. Toutefois, les structures des couches de protocole de communication des technologies actuelles de réseaux domestiques sont indépendantes, comme représenté à la Figure 1, avec des configurations d'adressage différentes, des configurations de transmission de données différentes, des méthodes de traitement de données différentes et des formats de données différents.

Pour résoudre les problèmes associés à l'interopérabilité et à l'interconnectivité entre des technologies de réseau hétérogènes, la présente norme vise à définir la couche CCP en tant que type de couche de convergence.

D'autres objectifs de la présente norme comportent la spécification de la méthode de transmission des données, la structure des paquets et le système d'adressage commun, ainsi

que le protocole de signalisation pour fournir des services de contrôle, A/V et de données de réseaux domestiques.

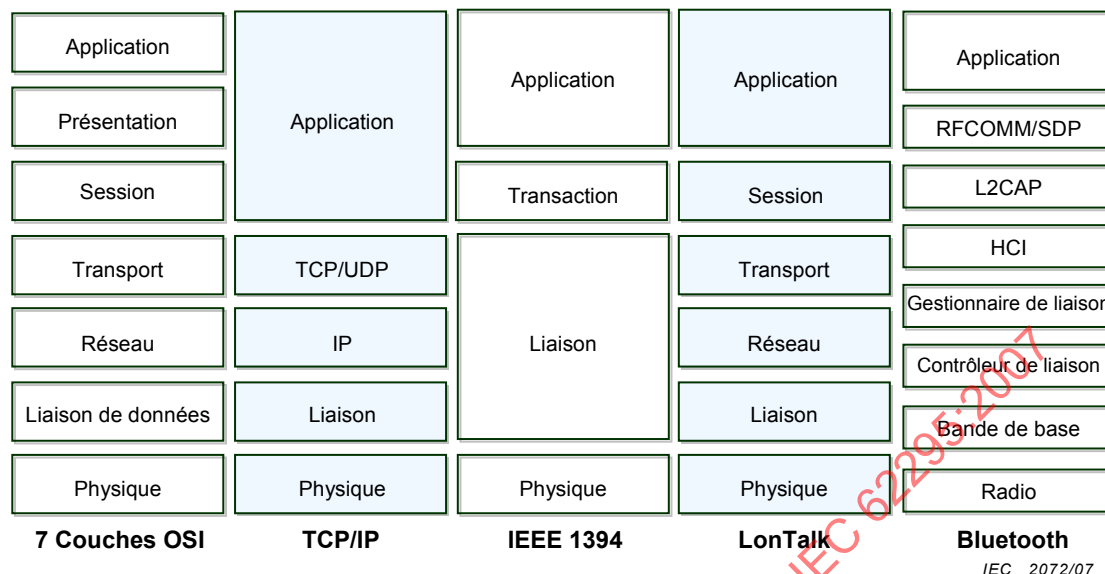


Figure 1 – Structures des couches de communication des technologies de réseau

## 2 Références normatives

Néant.

## 3 Termes, définitions, abréviations et conventions

Pour les besoins du présent document, les définitions suivantes s'appliquent.

### 3.1 Termes et définitions

#### 3.1.1

##### dispositif de CCP

dispositif ayant des capacités de mise en réseau utilisant les technologies de réseau disponibles dans le commerce pour liaison à un réseau domestique; il prend également en charge au moins le protocole de gestion de réseau domestique (HNMP) parmi quatre protocoles prévus dans la couche CCP et le CCP

#### 3.1.2

##### adressage de CCP

système d'adressage commun utilisé dans la couche CCP, constitué de quatre champs: domain address (adresse de domaine), cluster address (adresse de grappe) et device ID (identificateur de dispositif)

NOTE L'une des caractéristiques de l'adressage de CCP est qu'il possède une structure comprise par les programmes d'application incorporés dans des dispositifs liés à des réseaux hétérogènes, ainsi que par les utilisateurs des dispositifs.

#### 3.1.3

##### interface de serveur domestique

##### HSI (*home server interface*)

module d'interface situé dans un serveur domestique, responsable de la connexion avec un réseau particulier, pouvant traiter l'interface physique correspondante et le protocole de communication en fonction du type de réseau auquel il est connecté

NOTE Une interface de serveur domestique est représentée à la Figure 2.

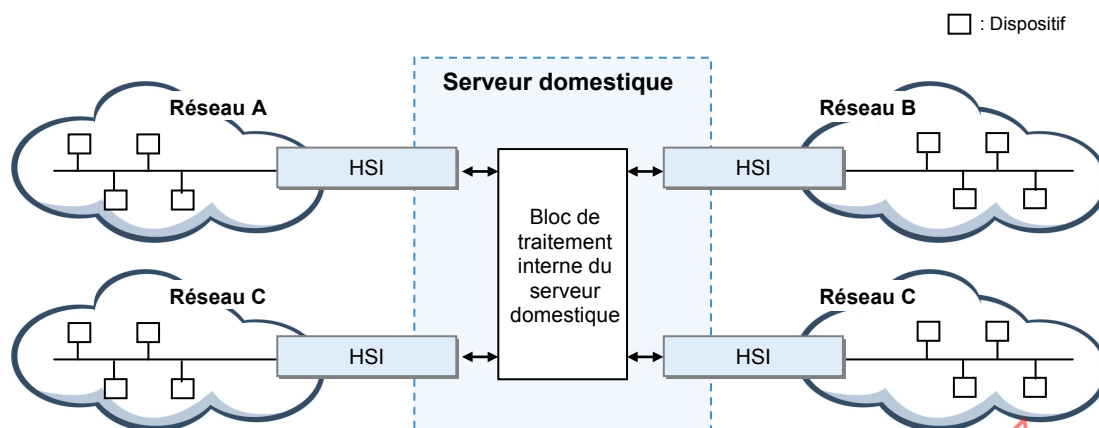


Figure 2 – Interface de serveur

### 3.1.4

#### réseau en grappe

collection de dispositifs utilisant une interface physique et un protocole de communication identiques

NOTE 1 Comme représenté à la Figure 3, un réseau en grappe est une collection de dispositifs utilisant une interface physique et un protocole de communication identiques. De plus, une interface de serveur domestique (HSI) reliant un serveur domestique aux dispositifs du réseau en grappe fait également partie du réseau en grappe. Une adresse de grappe est l'un des champs d'adressage de CCP définis dans la couche CCP, et elle est utilisée pour distinguer plusieurs réseaux en grappe reliés à un unique serveur domestique.

NOTE 2 Une interface de serveur domestique (HSI) reliant un serveur domestique aux dispositifs du réseau en grappe fait également partie du réseau en grappe.

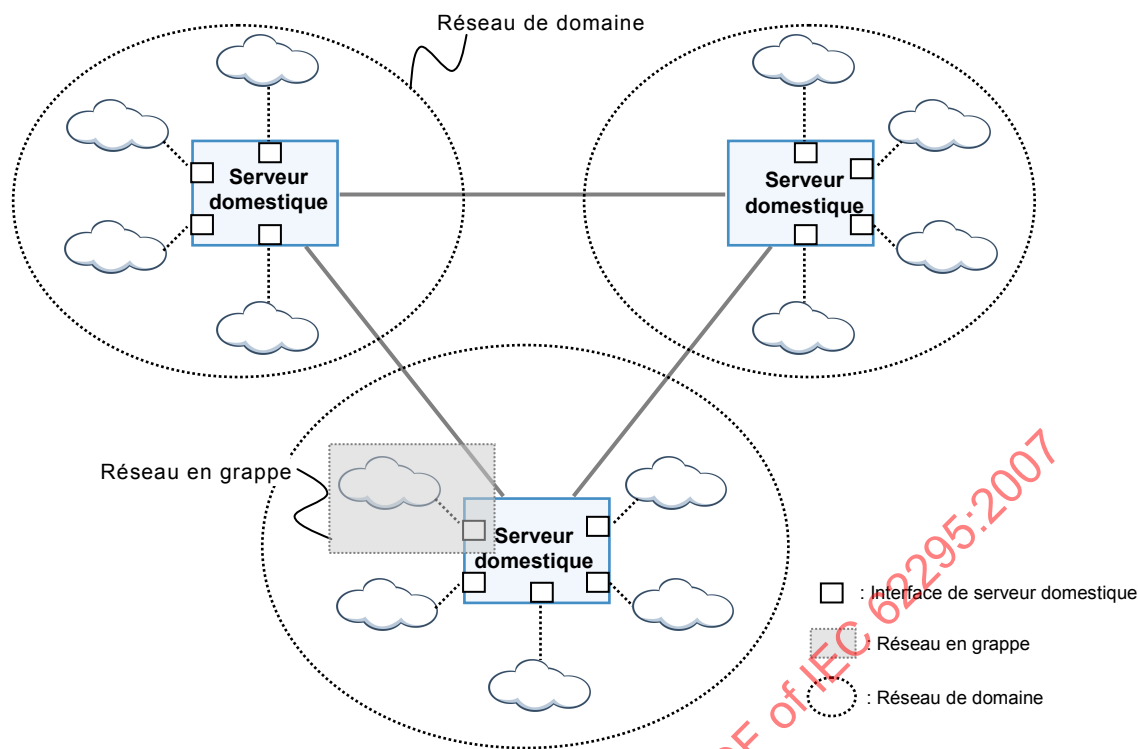
NOTE 3 Un réseau en grappe est représenté à la Figure 3.

### 3.1.5

#### adresse de grappe

l'un des champs d'adressage de CCP définis dans la couche CCP qui est utilisé pour distinguer plusieurs réseaux en grappe reliés à un unique serveur domestique

NOTE Une adresse de grappe est représentée à la Figure 3.



**Figure 3 – Réseau en grappe et réseau de domaine**

### 3.1.6

#### **réseau de domaine**

ensemble de dispositifs reliés à un unique serveur domestique

NOTE 1 Les dispositifs situés dans les réseaux en grappe multiples liés au serveur domestique et les interfaces de serveur domestique (HSI) gérant l'interface des réseaux en grappe constituent le réseau de domaine.

NOTE 2 Un réseau de domaine est représenté à la Figure 3.

### 3.1.7

#### **adresse de domaine**

l'un des champs d'adressage de CCP définis dans la couche CCP qui est utilisé pour distinguer des réseaux de domaine multiples au domicile

NOTE Une adresse de domaine est représentée à la Figure 3.

### 3.1.8

#### **réseau de serveur domestique (HS) unique**

réseau domestique comprenant un unique serveur domestique où il n'y a qu'un seul réseau de domaine dans l'environnement du réseau de serveur domestique (HS) unique

### 3.1.9

#### **réseau de serveurs domestiques (HS) multiples**

réseau domestique avec plusieurs serveurs domestiques où il y a plusieurs réseaux de domaine dans l'environnement du réseau de serveurs domestiques (HS) multiples

NOTE Un réseau de serveurs domestiques (HS) multiples est représenté à la Figure 3.

### 3.1.10

#### **programme d'application de CCP**

un programme d'application de CCP est placé en haut de la couche CCP. Un programme d'application de CCP envoie et reçoit des données vers/depuis la couche CCP par l'intermédiaire d'une interface de programmation d'application CCP

### 3.1.11

#### **unité de données de programme d'application**

##### **APDU (*application program data unit*)**

unité de données échangée entre plusieurs programmes d'application à un niveau équivalent

### 3.1.12

#### **interface de programmation (API, *application programming interface*) CCP**

interface entre le programme d'application et la couche CCP

### 3.1.13

#### **couches de protocole inférieures**

les couches de protocole inférieures se réfèrent à toutes les couches de protocole de communication au-dessous de la couche CCP, à l'exception de la couche physique

NOTE Un schéma des couches de protocole inférieures est représenté à la Figure 5.

### 3.1.14

#### **interface des couches de protocole inférieures**

interface entre l'APDU (Unité de données de programme d'application) et les couches de protocole inférieures, appelée interface des couches de protocole inférieures

NOTE Un schéma de l'interface des couches de protocole inférieures est représenté à la Figure 5.

## **3.2 Abréviations**

|            |                                                                                            |
|------------|--------------------------------------------------------------------------------------------|
| ACDS       | Service de contrôle actif et de données ( <i>Active Control &amp; Data Service</i> )       |
| ACMS       | Service de contrôle actif et multimédia ( <i>Active Control &amp; Multimedia Service</i> ) |
| ACS        | Service de contrôle actif ( <i>Active Control Service</i> )                                |
| ALLSTAT    | Tous les états ( <i>All Status</i> )                                                       |
| APDU       | Unité de données de programme d'application                                                |
| API        | Interface de programmation                                                                 |
| AT         | Type d'action ( <i>Action Type</i> )                                                       |
| ATTR       | Attribut                                                                                   |
| CCP        | Protocole de communication commun ( <i>Common Communication Protocol</i> )                 |
| CCPADDRVER | Version d'adresse CCP ( <i>CCP Address Version</i> )                                       |
| CCPDEV     | Dispositif de CCP ( <i>CCP Device</i> )                                                    |
| CCPH       | En-tête CCP ( <i>CCP Header</i> )                                                          |
| CCPHDRVER  | Version d'entête CCP ( <i>CCP Header Version</i> )                                         |
| CCPID      | Identification de CCP ( <i>CCP Identification</i> )                                        |
| CCPLEN     | Longueur de charge utile CCP ( <i>CCP Payload Length</i> )                                 |
| CMD        | Commande                                                                                   |
| CT         | Type de distribution ( <i>Cast Type</i> )                                                  |
| CTRLSTAT   | Control Status ( <i>Control Status</i> )                                                   |
| DATALEN    | Longueur des données ( <i>Data Length</i> )                                                |

|          |                                                                                           |
|----------|-------------------------------------------------------------------------------------------|
| DEV      | Dispositif ( <i>Device</i> )                                                              |
| DMS      | Service de données et multimédia ( <i>Data &amp; Multimedia Service</i> )                 |
| DS       | Service de données ( <i>Data Service</i> )                                                |
| DSTADDR  | Adresse destination ( <i>Destination Address</i> )                                        |
| ERRCODE  | Code d'erreur ( <i>Error Code</i> )                                                       |
| FMTS     | Taille finale de transfert maximale ( <i>Final Maximum Transfer Size</i> )                |
| FS       | Service complet ( <i>Full Service</i> )                                                   |
| FSM      | Machine à états finis ( <i>Finite State Machine</i> )                                     |
| HDSP     | Protocole de services de données domestiques ( <i>Home Data Service Protocol</i> )        |
| HDSPPLEN | Longueur de charge utile HDSP ( <i>HDSP Payload Length</i> )                              |
| HMSP     | Protocole de services multimédias domestiques ( <i>Home Multimedia Service Protocol</i> ) |
| HNMP     | Protocole de gestion de réseau domestique ( <i>Home Network Management Protocol</i> )     |
| HNMPCMD  | Commande HNMP ( <i>HNMP Command</i> )                                                     |
| HNMPPLEN | Longueur de charge utile HNMP ( <i>HNMP Payload Length</i> )                              |
| HS       | Serveur domestique ( <i>Home Server</i> )                                                 |
| HSI      | Interface de serveur domestique ( <i>Home Server Interface</i> )                          |
| MCS      | Service de contrôle de surveillance ( <i>Monitoring Control Service</i> )                 |
| MON      | Surveillance ( <i>Monitoring</i> )                                                        |
| MS       | Service multimédia ( <i>Multimedia Service</i> )                                          |
| MT       | Type de message ( <i>Message Type</i> )                                                   |
| MTS      | Taille de transfert maximale ( <i>Maximum Transfer Size</i> )                             |
| MYMTS    | Ma taille de transfert maximale ( <i>My Maximum Transfer Size</i> )                       |
| PATHLEN  | Longueur de chemin ( <i>Path Length</i> )                                                 |
| PCS      | Service de contrôle passif ( <i>Passive Control Service</i> )                             |
| PnP      | Prêt à l'emploi ( <i>Plug-and-Play</i> )                                                  |
| PT       | Type de charge utile ( <i>Payload Type</i> )                                              |
| QoS      | Qualité de service ( <i>Quality of Service</i> )                                          |
| REG      | Enregistrement ( <i>Registration</i> )                                                    |
| REGSTAT  | État d'enregistrement ( <i>Registration Status</i> )                                      |
| RES      | Réponse ( <i>Response</i> )                                                               |
| SEC      | Seconde                                                                                   |
| SEGNO    | Numéro de segmentation ( <i>Segmentation Number</i> )                                     |
| SEQNO    | Numéro de séquence ( <i>Sequence Number</i> )                                             |
| SRCADDR  | Adresse source ( <i>Source Address</i> )                                                  |
| STAT     | État ( <i>Status</i> )                                                                    |
| TID      | Identificateur de transaction ( <i>Transaction ID</i> )                                   |
| TT       | Type de trafic                                                                            |

|         |                                                                                          |
|---------|------------------------------------------------------------------------------------------|
| UHCP    | Protocole de contrôle domestique universel<br>( <i>Universal Home Control Protocol</i> ) |
| UHCPLEN | Longueur de charge utile UHCP ( <i>UHCP Payload Length</i> )                             |

### 3.3 Conventions

Le contenu de la charge utile UHCP est exprimé par des caractères en italique comme suit:

par exemple *<ITEM>argument</ITEM>*

## 4 Exigences

Cet article décrit les exigences de mise en œuvre des dispositifs interface de serveur domestique (HSI) et CCP.

### 4.1 Exigences concernant l'interface de serveur domestique

L'interface de serveur domestique (HSI) doit satisfaire aux exigences suivantes.

#### 4.1.1 Exigences de base pour la fourniture de données

- Pour prendre en charge le processus d'enregistrement de dispositif, l'interface de serveur domestique (HSI) doit être capable de recevoir tous les paquets diffusés vers le réseau en grappe depuis la couche physique et de les relayer à la couche CCP.
- L'interface de serveur domestique (HSI) doit maintenir et gérer une table d'adresses contenant les adresses CCP des dispositifs de CCP dans son réseau en grappe ainsi que les adresses physiques utilisées dans le réseau en grappe.
- À la fin du processus d'enregistrement d'un dispositif, lors de la fourniture des paquets à un dispositif de CCP particulier dans son réseau en grappe, l'interface de serveur domestique (HSI) doit être capable de transmettre les paquets provenant de la couche physique en utilisant l'adresse physique du dispositif de CCP qui correspond à l'adresse CCP destination.
- Une interface de serveur domestique (HSI) doit prendre en charge la couche CCP ainsi que les exigences spécifiées en 4.1. L'interface de serveur domestique (HSI) peut être mise en œuvre avec le type de module de carte pour l'interface physique, le protocole de communication et la couche CCP.

#### 4.1.2 Exigences fonctionnelles concernant le HNMP

- Après réception d'un paquet Device Registration Request (Requête d'enregistrement de dispositif) (DEV\_REG\_REQ), l'interface de serveur domestique (HSI) doit être capable d'envoyer un paquet Device Registration Response (Réponse à une requête d'enregistrement de dispositif) (DEV\_REG\_RES) au dispositif de CCP d'où provient le paquet DEV\_REG\_REQ.
- L'interface de serveur domestique (HSI) doit être capable de mémoriser dans la table d'adresses les adresses CCP et physique du dispositif de CCP, contenues dans le paquet DEV\_REG\_REQ.
- L'interface de serveur domestique (HSI) doit être capable d'allouer un Device ID (identificateur de dispositif) unique au dispositif de CCP d'où provient le paquet DEV\_REG\_REQ. L'interface de serveur domestique (HSI) doit également spécifier des données telles que le Device ID (Identificateur de dispositif) alloué, l'adresse de domaine et l'adresse physique propre de l'interface de serveur domestique (HSI) dans la charge utile HNMP du paquet DEV\_REG\_RES avant transmission.
- L'interface de serveur domestique (HSI) doit être capable de notifier qu'un nouveau dispositif de CCP a été enregistré dans le réseau domestique, à tous les dispositifs de CCP de son réseau en grappe ainsi qu'à toutes les autres interfaces de serveur

domestique (HSI) dans son réseau de domaine en transmettant un paquet Add Device (Ajout de dispositif) (ADD\_DEV).

- Pour prendre en charge la fonction Plug-and-Play (Prêt à l'emploi) (PnP) du dispositif de CCP, l'interface de serveur domestique (HSI) doit envoyer régulièrement un paquet Alive-Check Request (Requête de vérification d'activité) (ALV\_CHK\_REQ) aux dispositifs de CCP ayant achevé le processus d'enregistrement de dispositif et vérifier les paquets de réponse qui en proviennent. L'interface de serveur domestique (HSI) doit être capable de déterminer que les dispositifs de CCP ne répondant pas ont perdu la liaison avec le réseau domestique.
- L'interface de serveur domestique (HSI) doit être capable de notifier qu'un dispositif de CCP particulier a été déconnecté du réseau domestique, à tous les dispositifs de CCP de son réseau en grappe ainsi que toutes les autres HSI dans son réseau de domaine en transmettant un paquet Delete Device (Supprimer dispositif) (DEL\_DEV).

#### **4.1.3 Exigences concernant l'interface de serveur domestique pour monodiffusion, multidiffusion et diffusion générale**

- L'interface de serveur domestique (HSI) doit être capable de traiter tous les paquets reçus au niveau de sa couche physique, de la couche physique aux couches de protocole inférieures, avant de les envoyer à la couche CCP.
- L'interface de serveur domestique (HSI) doit être capable d'examiner l'adresse de grappe dans le champ CCP destination address (Adresse CCP destination) d'un paquet arrivant au niveau de la couche CCP par l'intermédiaire des couches de protocole inférieures et d'envoyer le paquet à l'interface de serveur domestique (HSI) responsable de l'interface du réseau en grappe correspondant à l'adresse de grappe.
- Lorsqu'un paquet CCP arrive d'une autre interface de serveur domestique (HSI)
  - si le type de diffusion du paquet CCP reçu est monodiffusion et que l'adresse CCP destination est spécifiée, alors l'interface de serveur domestique (HSI) recevant le paquet doit être capable de relayer le paquet au dispositif de CCP particulier situé dans son réseau en grappe,
  - si le type de diffusion du paquet CCP reçu est multidiffusion, alors l'interface de serveur domestique (HSI) recevant le paquet doit être capable de relayer le paquet uniquement vers le nombre limité de dispositifs de CCP situés dans son réseau en grappe en fonction du contenu de la multidiffusion,
  - si le type de diffusion du paquet CCP reçu est diffusion générale, alors l'interface de serveur domestique (HSI) recevant le paquet doit être capable de relayer le paquet à tous les dispositifs de CCP situés dans son réseau en grappe.

#### **4.2 Exigences concernant le dispositif de CCP**

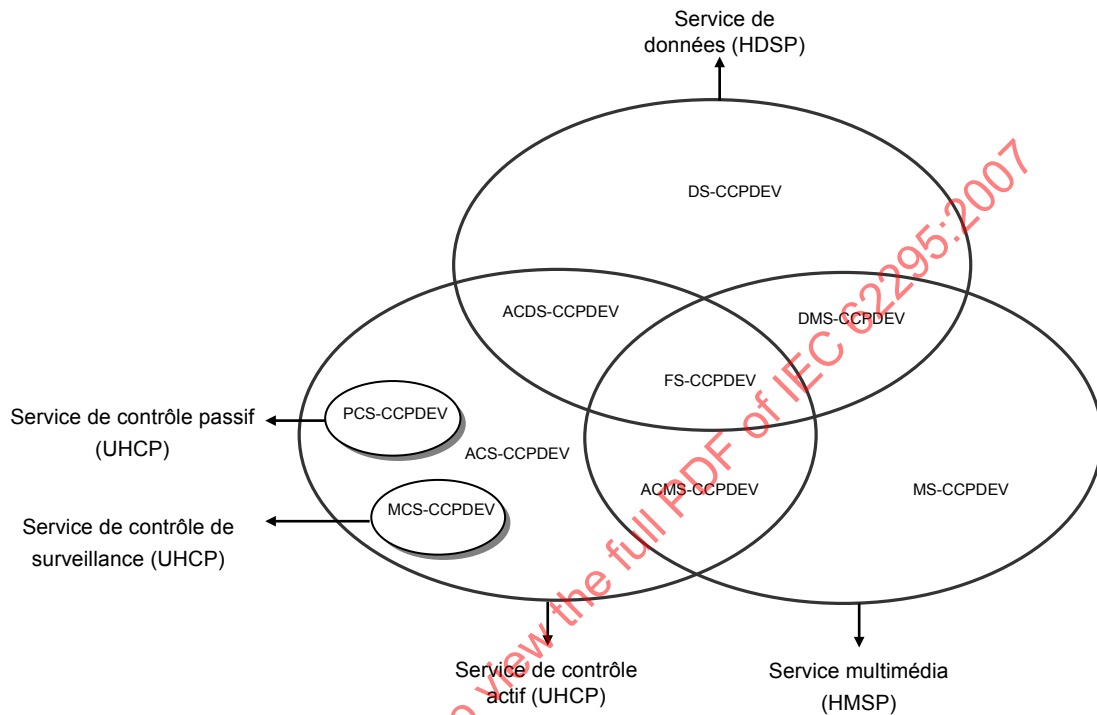
Ce paragraphe spécifie les exigences concernant le processus d'enregistrement de dispositif, nécessaires pour que les dispositifs de CCP exécutent la fonction « Prêt à l'emploi », ainsi que les exigences concernant la classification des dispositifs de CCP selon les fonctions CCP assurées par les dispositifs de CCP.

##### **4.2.1 Exigences concernant l'enregistrement de dispositif**

- Si un dispositif de CCP n'est pas informé de l'adresse physique de l'interface de serveur domestique (HSI) de son propre réseau en grappe, l'adresse physique (ou l'adresse de réseau) de l'interface de serveur domestique (HSI) doit être obtenue par l'intermédiaire du processus d'enregistrement de dispositif.
- Pour les besoins de l'enregistrement de dispositif, un dispositif de CCP doit être capable de diffuser initialement un paquet device registration request (requête d'enregistrement de dispositif) (DEV\_REG\_REQ) à son propre réseau en grappe.
- À la fin du processus d'enregistrement de dispositif, lorsqu'un dispositif de CCP transmet des données à un réseau hétérogène, il doit envoyer tous les paquets de la couche physique à l'interface de serveur domestique (HSI) de son propre réseau en grappe, quel que soit le type de diffusion et l'adresse CCP destination.

#### 4.2.2 Exigences concernant la classification des dispositifs de CCP

Les dispositifs de CCP se réfèrent à des dispositifs prenant en charge HNMP parmi les fonctions fournies par la couche CCP et le CCP. Les dispositifs de CCP peuvent être classés en fonction de leurs capacités de traitement et des technologies de réseau utilisées. La présente norme doit classer les dispositifs de CCP en neuf catégories, selon les fonctions fournies par les dispositifs de CCP, comme représenté à la Figure 4.



IEC 2075/07

**Figure 4 – Classification des dispositifs de CCP**

- Un dispositif de CCP Active Control Service (Service de contrôle actif) (ACS-CCPDEV) est un dispositif de CCP prenant uniquement en charge la fonction UHCP.
- Un dispositif de CCP Passive Control Service (Service de contrôle passif) (PCS-CCPDEV) est un dispositif de CCP dont la fonction UHCP est exécutée en acceptant des commandes provenant d'autres dispositifs.
- Un dispositif de CCP Monitoring Control Service (Service de contrôle de surveillance) (MCS-CCPDEV) est un dispositif de CCP assurant uniquement les capacités de surveillance parmi les fonctions de l'UHCP.
- Un dispositif de CCP Data Service (Service de données) (DS-CCPDEV) est un dispositif de CCP fournissant uniquement le HDSP.
- Un dispositif de CCP Multimedia Service (Service multimédia) (MS-CCPDEV) est un dispositif de CCP assurant uniquement la fonction HMSP.
- Un dispositif de CCP Active Control & Data Service (Service de contrôle actif et de données) (ACDS-CCPDEV) est un dispositif de CCP assurant les fonctions UHCP et HDSP.
- Un dispositif de CCP Active Control & Multimedia Service (Service de contrôle actif et multimédia) (ACMS-CCPDEV) est un dispositif de CCP fournissant l'UHCP et le HMSP.
- Un dispositif de CCP Data & Multimedia Service (Service de données et multimédia) (DMS-CCPDEV) est un dispositif de CCP fournissant le HDSP et le HMSP.

- Un dispositif de CCP Full Service (Service complet) (FS-CCPDEV) est un dispositif de CCP assurant les fonctions UHCP, HDSP et HMSP, et il est également équipé d'un panneau d'affichage pour les fonctions d'affichage.

## **5 Couche de protocole de communication commun (CCP)**

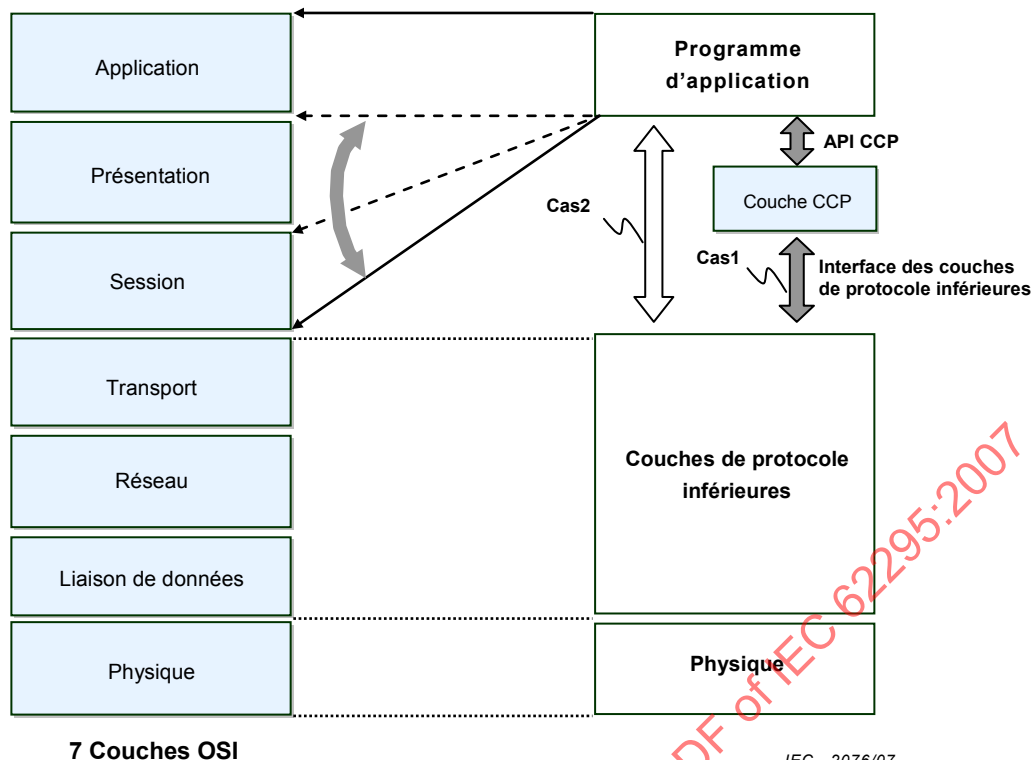
Cet article spécifie la couche CCP permettant une communication bilatérale entre des réseaux hétérogènes de divers types de programmes d'application.

### **5.1 Couche CCP**

La couche CCP fournit un canal de communication commun pour des dispositifs (ou des applications exécutées depuis les dispositifs) reliés à des réseaux hétérogènes. La couche CCP doit être située au-dessous de tous les programmes d'application fournis par les fournisseurs de service (ou les fabricants de dispositif). Du point de vue des 7 couches OSI, bien que la plupart des protocoles de réseaux et des programmes d'application soient similaires pour la structure OSI à 7 couches, ils ne sont pas divisés nettement en sept couches. L'emplacement de la couche CCP est variable selon la couche couverte par le programme d'application (ou la structure des couches de protocole inférieures).

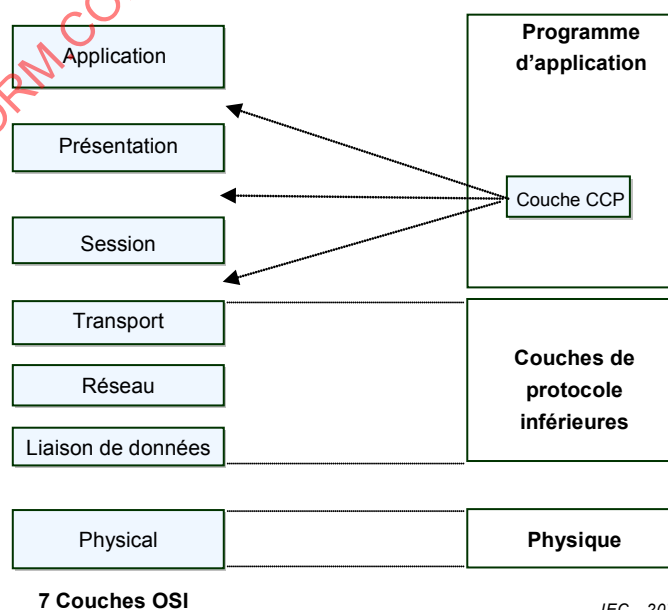
Le Cas 1 de la Figure 5 est un exemple où un programme d'application d'un dispositif de réseaux domestiques envoie des données à des couches de protocole inférieures par l'intermédiaire de la couche CCP, tandis que le Cas 2 présente un exemple de programme d'application d'un dispositif de réseau domestique envoyant directement des données à des couches de protocole inférieures sans passer par la couche CCP.

Le programme d'application du cas 1 transmet une Unité de données de programme d'application (APDU) désirée à un dispositif connecté à un réseau en grappe lié à un réseau hétérogène par l'intermédiaire du serveur domestique. Par ailleurs, l'exemple du Cas 2 peut être utilisé lorsque le programme d'application doit envoyer une Unité de données de programme d'application (APDU) désirée à un dispositif situé dans le même réseau en grappe en utilisant le même protocole de réseau, en contournant le serveur domestique. En conséquence, dans le Cas 2, tous les services ayant été proposés dans le même réseau avant d'appliquer la couche CCP (par exemple, FTP sur les réseaux TCP/IP et le service Telnet) restent disponibles.



**Figure 5 – Définitions du programme d'application, interface de programmation (API) CCP, interface des couches de protocole inférieures et couches de protocole inférieures**

Dans le cas d'un programme d'application ne couvrant que jusqu'à la couche application comme représenté à la Figure 6, la couche CCP est située au-dessous de la couche application. Pour un programme d'application couvrant de la couche application à la couche présentation, la couche CCP est située au-dessous de la couche présentation. Pour un programme d'application couvrant de la couche application à la couche session, la couche CCP est située au-dessous de la couche session.



**Figure 6 – Emplacement de la couche CCP**

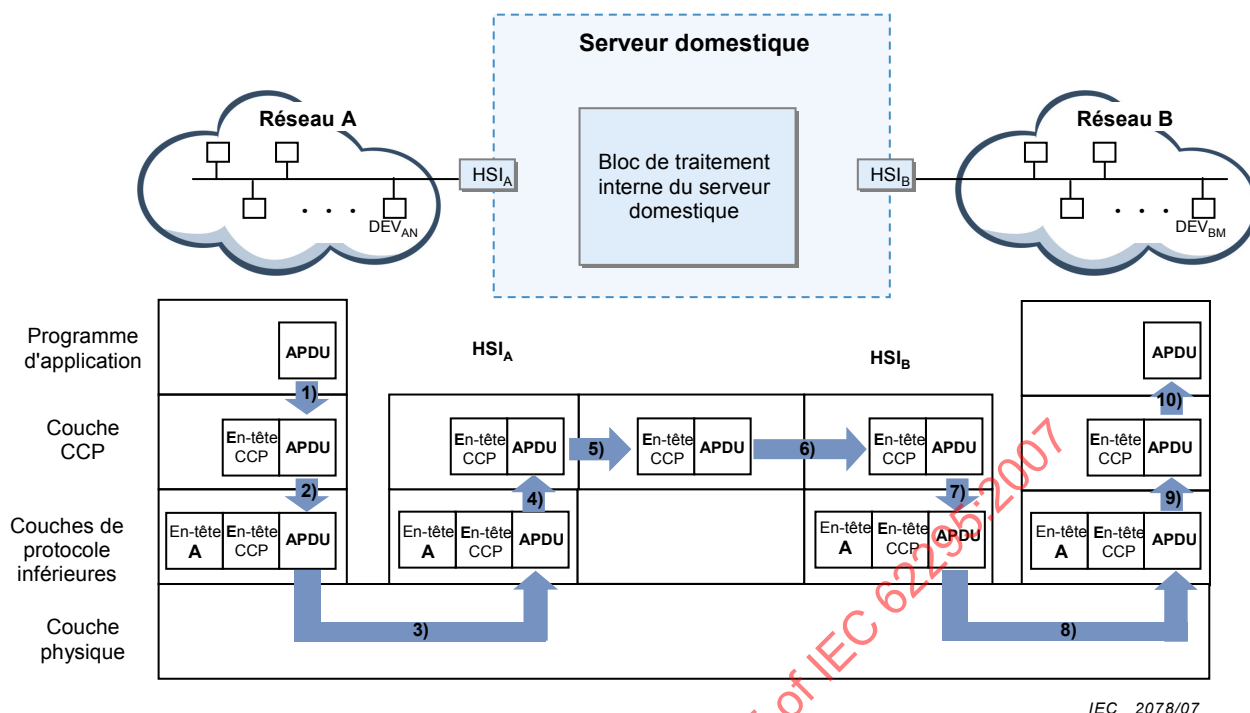
## 5.2 Fourniture de données sur des réseaux hétérogènes

Ce paragraphe décrit le processus de transmission de données en utilisant la Figure 7. Pour la communication de données avec un réseau hétérogène, une unité de données de programme d'application (APDU) doit passer par la couche CCP avant d'être envoyée aux couches de protocole inférieures. Pour qu'un dispositif de réseau domestique transmette une Unité de données de programme d'application (APDU) à un réseau hétérogène, le programme d'application envoie une APDU à la couche CCP par l'intermédiaire de l'Interface de programmation (API) CCP et la couche CCP encapsule l'APDU reçue avec un en-tête CCP et la retransmet aux couches de protocole inférieures.

- DEV<sub>An</sub>: Dispositif de CCP dans le Réseau A avec Device ID (Identificateur de dispositif) n
- DEV<sub>Bm</sub>: Dispositif de CCP dans le Réseau B avec Device ID (Identificateur de dispositif) m
- HSI<sub>A</sub>, HSI<sub>B</sub>: Interface de serveur domestique (HSI) responsable de l'interface avec le Réseau A (ou le Réseau B) dans le serveur domestique.

Le processus de transmission de données est le suivant.

- 1) Le programme d'application de DEV<sub>An</sub> envoie une unité de données de programme d'application (APDU) à la couche CCP par l'intermédiaire de l'Interface de programmation (API) CCP.
- 2) La couche CCP de DEV<sub>An</sub> encapsule l'unité de données de programme d'application (APDU) provenant du programme d'application avec un en-tête CCP et l'envoie aux couches de protocole inférieures par l'intermédiaire de l'interface de couches de protocole inférieures.
- 3) Les couches de protocole inférieures génèrent un en-tête pour chaque couche correspondante et la couche physique désigne l'adresse physique de HSI<sub>A</sub> comme adresse destination et transmet le paquet.
- 4) La couche physique de HSI<sub>A</sub> reçoit le paquet et le retransmet aux couches de protocole inférieures. Les couches de protocole inférieur traitent un en-tête pour chaque couche correspondante et l'envoient à la couche CCP.
- 5) La couche CCP de HSI<sub>A</sub> retransmet le paquet CCP qui est arrivé au bloc de traitement interne du serveur domestique.
- 6) Le bloc de traitement interne du serveur domestique analyse les informations d'en-tête CCP du paquet reçu pour traiter le paquet et l'envoie à l'interface de serveur domestique (HSI) (HSI<sub>B</sub>) correspondant à l'adresse de grappe de l'adresse CCP destination.
- 7) La couche CCP de HSI<sub>B</sub> transmet le paquet reçu aux couches de protocole inférieures. Ce faisant, HSI<sub>B</sub> doit chercher et retrouver dans la table d'adresses l'adresse physique de DEV<sub>Bm</sub> correspondant à l'adresse CCP destination et la notifier à la couche physique.
- 8) La couche physique de HSI<sub>B</sub> désigne l'adresse physique de DEV<sub>Bm</sub> comme adresse destination et transmet le paquet.
- 9) La couche physique de DEV<sub>Bm</sub> reçoit le paquet et l'envoie aux couches de protocole inférieures où chaque en-tête est traité en correspondance avec chaque couche et le paquet est transmis à la couche CCP.
- 10) La couche CCP de DEV<sub>Bm</sub> délivre le paquet qui est arrivé au programme d'application par l'intermédiaire de l'Interface de programmation (API) CCP.



IEC 2078/07

**Figure 7 – Exemple de transmission de données sur des réseaux hétérogènes utilisant la couche CCP**

## 6 Adressage de CCP

### 6.1 Généralités

Pour mettre en œuvre l'intercommunication et l'interopérabilité dans l'environnement de réseau domestique où coexistent divers réseaux hétérogènes, les dispositifs constituant le réseau domestique doivent être vus les uns les autres comme étant liés à un réseau unique, quel que soit le type de réseau auquel chaque dispositif est connecté. Cette idée ne peut être matérialisée que s'il existe une configuration d'adressage pouvant être comprise entre les dispositifs des différents réseaux. De plus, une configuration d'adressage pouvant être comprise par des êtres humains et des programmes d'application facilitera son utilisation et facilitera la mise en œuvre des programmes d'application.

Cet article spécifie une nouvelle configuration d'adressage de CCP conforme aux trois exigences fonctionnelles suivantes:

- Une nouvelle structure d'adresse commune adéquate pour l'environnement de réseau domestique.
- Un mécanisme d'adressage pouvant être compris par l'utilisateur ainsi que par le programme d'application.

### 6.2 Structure d'adressage facilitant la commutation du trafic pour le serveur domestique. Version 0

Ce paragraphe décrit le format d'adresse CCP de la version 0. Le champ version concernant l'adressage de CCP est décrit à l'Article 8. Le format d'adresse CCP peut être modifié ou étendu pour utilisation future, comme les autres protocoles ayant leur propre configuration d'adressage. Dans tous les cas, tous les dispositifs de CCP d'adresse CCP version 0 doivent satisfaire aux exigences décrites en 7.2.

Pour satisfaire aux exigences fonctionnelles, le format d'adresse CCP version 0 est défini par quatre octets, comme représenté à la Figure 8. L'adresse CCP contient trois champs: Domain

Address (adresse de domaine), Cluster Address (adresse de grappe) et Device ID (Identificateur de dispositif).

|                                    |                                   |                                            |
|------------------------------------|-----------------------------------|--------------------------------------------|
| Adresse<br>de domaine<br>(1 octet) | Adresse<br>de grappe<br>(1 octet) | Identificateur de dispositif<br>(2 octets) |
|------------------------------------|-----------------------------------|--------------------------------------------|

**Figure 8 – Format d'adresse CCP d'une adresse CCP version 0**

### 6.2.1 Domain address (Adresse de domaine)

L'adresse de domaine distingue le réseau de domaine. Lorsqu'un réseau domestique situé dans un bureau (ou à domicile) est constitué de plusieurs serveurs domestiques, l'adresse de domaine est utilisée pour les différencier. Puisqu'un réseau de domaine constitue un serveur domestique unique, l'adresse de domaine peut être appelée home server ID (identificateur de serveur domestique).

L'adresse de domaine est constituée d'un octet. Il peut donc y avoir 256 adresses de domaine (de 0 à 255). Dans l'environnement de réseau de Serveur domestique (HS) unique, on doit utiliser une adresse de domaine égale à 1. Puisqu'un dispositif ne connaît pas sa propre adresse de domaine avant d'avoir terminé le processus d'enregistrement de dispositif décrit en 9.3, le dispositif doit utiliser initialement l'adresse de domaine « 0 ».

Dans l'environnement de réseau de Serveurs domestiques (HS) multiples où il existe plusieurs serveurs domestiques, un processus est nécessaire pour allouer une adresse de domaine à chaque serveur domestique par l'intermédiaire de la procédure d'enregistrement de serveur domestique HNMP. Toutefois, cette version de document ne tient pas compte des normes d'enregistrement de serveur domestique dans l'environnement de réseau de Serveurs domestiques (HS) multiples.

### 6.2.2 Cluster address (Adresse de grappe)

L'adresse de grappe distingue plusieurs réseaux en grappe liés à un unique réseau de domaine. Puisqu'un seul réseau en grappe est lié à une seule interface de serveur domestique (HSI), l'adresse de grappe peut être utilisée pour commuter les paquets reçus dans le serveur domestique. Pour représenter l'adresse de grappe par une valeur unique, l'adresse de grappe doit être définie dans la présente norme comme le numéro de port d'une interface de serveur domestique (HSI). Il peut donc également y avoir 256 adresses de grappe (de 0 à 255). Puisqu'un dispositif ne connaît pas sa propre adresse de grappe avant d'avoir terminé le processus d'enregistrement de dispositif décrit en 9.3, le dispositif doit utiliser initialement l'adresse de grappe « 0 ».

### 6.2.3 Champ Device ID (Identificateur de dispositif)

Un champ device ID (identificateur de dispositif) de dispositif est utilisé pour distinguer des dispositifs de CCP ayant la même adresse de domaine et de grappe. Initialement, un dispositif de CCP n'est pas informé de son identificateur de dispositif. À chaque dispositif de CCP doit donc être alloué son identificateur de dispositif par l'interface de serveur domestique (HSI) par l'intermédiaire du processus d'enregistrement de dispositif, qui est l'une des fonctions du HNMP.

Lors de l'exécution du processus d'enregistrement de dispositif, l'interface de serveur domestique (HSI) doit être capable d'assigner différents identificateurs de dispositif à des dispositifs de CCP avec une adresse de domaine et de grappe identique.

L'identificateur de dispositif est un champ de deux octets. La plage d'identificateurs de dispositif va donc de 0 à 65 535. Avant d'achever le processus d'enregistrement de dispositif, un dispositif doit initialement utiliser l'identificateur de dispositif « 0 » et l'interface de serveur

domestique (HSI) doit assigner les identificateurs de dispositif de 1 à 65 535 en séquence, à chaque fois que des dispositifs s'enregistrent.

## 7 Format et champs des paquets CCP

### 7.1 Généralités

Cet article décrit la structure des paquets CCP utilisés dans la couche CCP ainsi que chaque champ utilisé dans le paquet CCP.

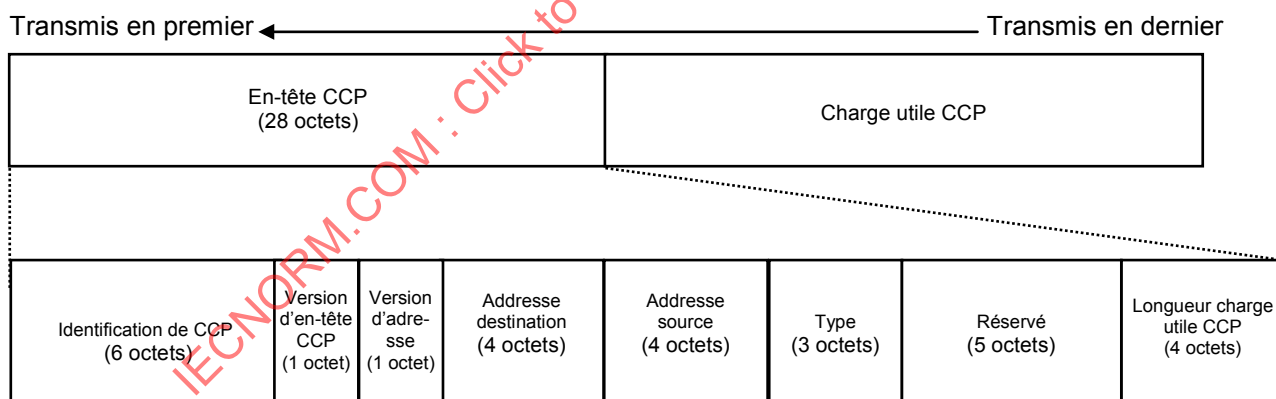
Le CCP fournit le Protocole de gestion de réseau domestique (HNMP) pour les services de gestion de base des réseaux domestiques, le Protocole de contrôle domestique universel (UHCP) pour les services de contrôle et de surveillance de dispositif, le Protocole de service de données domestiques (HDSP) pour les services de données et de messagerie et le Protocole de service multimédia domestique (HMSP) pour les services multimédias.

L'ensemble des protocoles mentionnés ci-dessus (à l'exception d'UHCP) sont des protocoles de signalisation basés sur les messages définis dans cette règle.

### 7.2 Format des paquets CCP

La Figure 9 représente le format des paquets CCP dans la version 0. Un paquet CCP est constitué principalement d'un en-tête CCP et d'une charge utile CCP. L'en-tête CCP a une longueur de 28 octets, contenant l'identification de CCP, l'adresse destination, l'adresse source, le champ Type, un champ réservé et un champ CCP Payload Length (longueur de charge utile CCP).

La charge utile CCP comprend les unités de données de programme d'application (APDU) reçues depuis le programme d'application.



IEC 2079/07

**Figure 9 – Format des en-têtes de paquet CCP, version 0.**

#### 7.2.1 CCP identification (Identification de CCP) (CCPID)

L'identification de CCP (CCPID) est utilisée pour déterminer si un paquet reçu au niveau de la couche CCP est ou non un paquet CCP.

S'il s'agit d'un paquet CCP, son en-tête CCP est traité et la charge utile CCP est envoyée au programme d'application. S'il ne s'agit pas d'un paquet CCP, le paquet reçu depuis les couches de protocole inférieures est directement transmis au programme d'application.

Le dispositif de CCP peut désormais communiquer avec d'autres dispositifs de CCP sur des réseaux hétérogènes ainsi qu'avec ceux qui se trouvent sur le même réseau en grappe.

CCPID a une longueur de six octets. Le champ CCPID est défini par une chaîne textuelle « IECccp », signifiant un en-tête CCP en tant que valeur de code ASCII (0x494543636370). La couche CCP doit initialement vérifier les quatre premiers octets d'un paquet reçu depuis les couches de protocole inférieures afin d'évaluer s'il s'agit d'un paquet CCP.

### 7.2.2 CCP header version (Version de l'en-tête CCP) (CCPHDRVER)

Puisque chaque champ de l'en-tête CCP peut être modifié ou étendu pour utilisation future, CCPHDRVER est défini après CCPID. La longueur du champ CCPHDRVER est d'un octet et il doit être mis à 0x00 dans le cas de la version 0. Ce numéro de version est incrémenté d'une unité s'il existe une quelconque modification ou variation du format et de la taille de l'en-tête CCP.

### 7.2.3 CCP address version (Version de l'adresse CCP) (CCPADDRVER)

Puisque chaque champ de l'adresse CCP peut être modifié ou étendu pour utilisation future, CCPADDRVER est défini après CCPHDRVER. La longueur du champ CCPADDRVER est d'un octet et il doit être mis à 0x00 dans le cas de la version 0. Ce numéro de version est également incrémenté d'une unité s'il existe une quelconque modification ou variation du format et de la taille de l'adresse CCP.

### 7.2.4 Destination address (Adresse destination) (DESTADDR)

DADDR indique l'adresse CCP du dispositif de CCP destination auquel doit être envoyé le paquet CCP. Si CCPADDRVER est à 0, l'adresse destination a une longueur de quatre octets.

Comme spécifié dans cet article, une adresse destination est constituée de trois champs: Domain Address (adresse de domaine), Cluster Address (adresse de grappe) et Device ID (identificateur de dispositif).

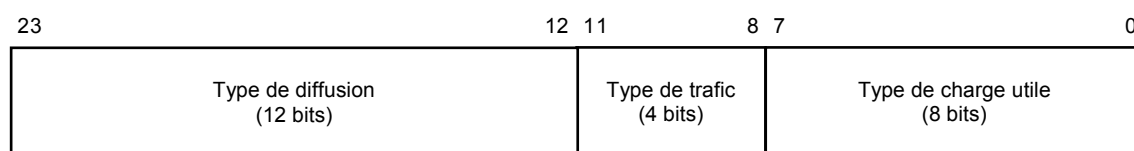
### 7.2.5 Source address (Adresse source) (SRCADDR)

SADDR indique l'adresse CCP du dispositif de CCP transmettant le paquet. Si CCPADDRVER est à 0, l'adresse source a une longueur de quatre octets et elle est également constituée de trois champs: Domain Address (adresse de domaine), Cluster Address (adresse de grappe) et Device ID (identificateur de dispositif).

### 7.2.6 Champ Type (TYPE)

Les champs type sont constitués de trois champs indiquant la méthode de fourniture des paquets, le type de trafic et le type de charge utile: Cast Type, Traffic Type et Payload Type. Les champs Type ont une longueur de trois octets.

La Figure 10 représente les champs Type.



IEC 2080/07

**Figure 10 – Champs Type**

#### 7.2.6.1 Cast type (Type de diffusion) (CT)

Le champ type de diffusion définit la façon dont le dispositif de CCP source délivre un paquet par l'intermédiaire du serveur domestique. C'est un champ de 12 bits.

Comme indiqué dans le Tableau 1, la présente norme fournit sept méthodes: Unicast (monodiffusion), Cluster-Multicast (multidiffusion de grappe), Domain-Multicast (multidiffusion de domaine), Device-Multicast (multidiffusion de dispositif), Device-Broadcast (diffusion générale), HS-Broadcast (diffusion générale de réseau domestique) et Broadcast (diffusion générale).

**Tableau 1 – Champ Cast type (Type de diffusion)**

| Type de diffusion[11:0]                                                                                   | Notation           | Description                                                                                                                                                                                                                                                                                                    |
|-----------------------------------------------------------------------------------------------------------|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x0XX <sup>a</sup>                                                                                        | Unicast            | Utilisé pour les communications 1:1 entre dispositifs de CCP.                                                                                                                                                                                                                                                  |
| 0x1WW <sup>b</sup>                                                                                        | Domain-Multicast   | Utilisé pour transmettre des paquets aux interfaces de serveur domestique (HSI), aux serveurs domestiques et à tous les dispositifs de CCP dans un réseau de domaine avec l'adresse de domaine WW.                                                                                                             |
| 0x2YY <sup>c</sup>                                                                                        | Cluster-Multicast  | Utilisé pour transmettre des paquets à tous les dispositifs de CCP incluant les interfaces de serveur domestique (HSI) dans le réseau de grappe avec l'adresse de grappe YY.                                                                                                                                   |
| 0xF0F                                                                                                     | Device-Broadcast   | Utilisé lorsqu'il y a plusieurs réseaux de domaine, par exemple dans l'environnement de réseau à plusieurs serveurs domestiques (HS), pour transmettre des paquets uniquement aux dispositifs de CCP situés dans le réseau de domaine.                                                                         |
| 0xFF0                                                                                                     | HS-Broadcast       | Utilisé lorsqu'il y a plusieurs réseaux de domaine, par exemple dans l'environnement de réseau à plusieurs serveurs domestiques (HS), pour transmettre des paquets à tous les serveurs domestiques et interfaces de serveur domestique (HSI) situés dans le réseau de domaine mais pas aux dispositifs de CCP. |
| 0xFFF                                                                                                     | Broadcast          | Utilisé lorsqu'il y a plusieurs réseaux de domaine, par exemple dans l'environnement de réseau à plusieurs serveurs domestiques (HS), pour transmettre des paquets à toutes les interfaces de serveur domestique (HSI), serveurs domestiques et dispositifs de CCP situés dans le réseau de domaine.           |
| Autres                                                                                                    | reserved (réserve) | Réservée pour de futures utilisations.                                                                                                                                                                                                                                                                         |
| <sup>a</sup> XX: Importe peu<br><sup>b</sup> WW: Adresse de domaine<br><sup>c</sup> YY: Adresse de grappe |                    |                                                                                                                                                                                                                                                                                                                |

#### 7.2.6.2 Traffic type (Type de trafic) (TT)

Le champ traffic type (type de trafic) est utilisé pour spécifier le type de données pour que le serveur domestique traite les paquets de manière interne lorsque la fonction Qualité de service (QoS) doit être traitée par le bloc de traitement interne du serveur domestique.

Le programme d'application peut configurer le type de trafic de l'en-tête CCP en vérifiant les caractéristiques de l'unité de données de programme d'application (APDU). Le serveur domestique peut ou non être conçu pour prendre en charge le contrôle de la Qualité de service (QoS) selon la valeur du champ type de trafic.

La présente norme ne spécifie pas s'il convient que le serveur domestique prenne en charge la Qualité de service (QoS) ou l'algorithme de prise en charge de la QoS.

Comme représenté dans le Tableau 2, le champ type de trafic est constitué de quatre bits.

**Tableau 2 – Champ Traffic type (Type de trafic)**

| Type de trafic[3:0] | Notation                                       | Description                                                                                                                                                         |
|---------------------|------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0000                | Importe peu                                    | Utilisé lorsqu'il n'est pas nécessaire de spécifier le type de trafic et que tout traitement de Qualité de service (QoS) est acceptable dans le serveur domestique. |
| 0100                | Control Data (Données de contrôle)             | Utilisé lorsque l'unité de données de programme d'application (APDU) est un message de signalisation pour gérer ou contrôler HNMP, UHCP, HDSP ou HMSP.              |
| 1000                | Non Real-Time Data (Données non en temps réel) | Utilisé pour des données générales, par exemple Internet.                                                                                                           |
| 1100                | Real-Time Data (Données en temps réel)         | Utilisé pour des données en temps réel telles que des flux audio ou vidéo.                                                                                          |
| autres              | reserved (réservé)                             | Réservé pour une future utilisation.                                                                                                                                |

### 7.2.6.3 Payload type (Type de charge utile) (PT)

Le champ payload type (type de charge utile) indique le type de charge utile CCP.

Des spécifications existent actuellement en ce qui concerne HNMP, UHCP, HDSP et HMSP, définis dans le CCP. Dans l'avenir, le champ type de charge utile pourra être utilisé dans des protocoles élaborés pour d'autres services utilisant le CCP. Comme représenté dans le Tableau 3, le champ type de charge utile est constitué d'un octet.

**Tableau 3 – Champ Payload type (Type de charge utile)**

| Type de charge utile[7:0] | Notation                                   | Description                                                                |
|---------------------------|--------------------------------------------|----------------------------------------------------------------------------|
| 0x00                      | General application (Application générale) | Indique que le type de charge utile fait partie de l'application générale. |
| 0x01                      | HNMP                                       | Indique que le type de charge utile est un paquet HNMP défini dans le CCP. |
| 0x02                      | UHCP                                       | Indique que le type de charge utile est un paquet UHCP défini dans le CCP. |
| 0x03                      | HDSP                                       | Indique que le type de charge utile est un paquet HDSP défini dans le CCP. |
| 0x04                      | HMSP                                       | Indique que le type de charge utile est un paquet HMSP défini dans le CCP. |
| autres                    | reserved (réservé)                         | Réservé pour une future utilisation.                                       |

### 7.2.7 Champ Reserved (Réservé) (RSV)

Le champ reserved (réservé) de cinq octets est réservé pour une future utilisation.

### 7.2.8 Champ CCP payload length (Longueur de charge utile CCP) (CCPPLEN)

Le champ CCP payload length (longueur de charge utile CCP) de quatre octets indique la taille en octets des unités de données de programme d'application (APDU) dans la charge utile.

### 7.2.9 Champ CCP payload (Charge utile CCP)

Le champ CCP payload (charge utile CCP) comprend les unités de données de programme d'application (APDU) générées par le programme d'application. La charge utile CCP est écrite avec les messages de signalisation des HMSP, HNMP, UHCP et HDSP, définis dans le CCP.

## 8 Protocole de gestion de réseau domestique (HNMP)

### 8.1 Généralités

Le HNMP est utilisé par le CCP pour exécuter des fonctions de gestion du réseau domestique. De plus, le HNMP gère les ressources du réseau domestique et fournit des fonctions Plug-and-play (Prêt à l'emploi) (PnP) pour le serveur domestique ainsi que les dispositifs de CCP.

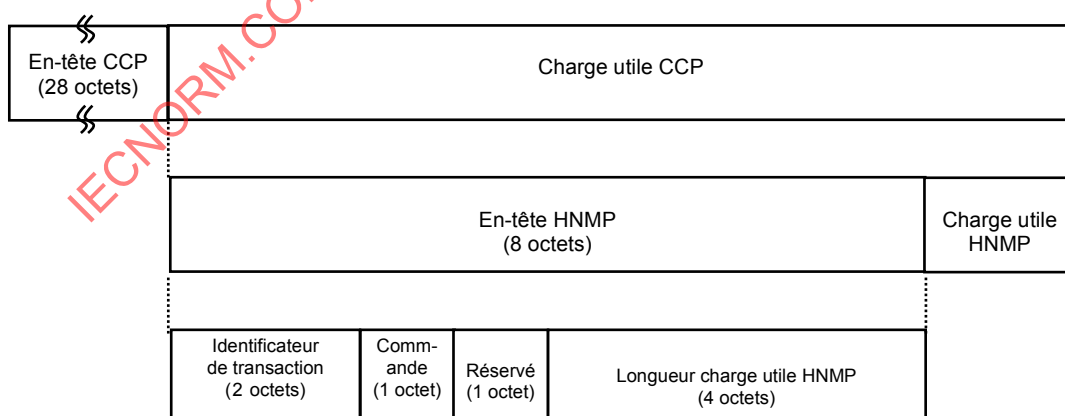
Le HNMP est un protocole de signalisation de base par paquets fourni par le CCP. De simples paquets de requête et de réponse sont définis dans le HNMP.

Tous les dispositifs de CCP sous l'environnement du réseau domestique constitué de réseaux hétérogènes doivent au moins prendre en charge la fonction d'enregistrement de dispositif, parmi les fonctions du HNMP. D'autres fonctions du HNMP peuvent être mises en œuvre de façon sélective selon la capacité de traitement et les domaines d'application des dispositifs de CCP.

### 8.2 Format des paquets HNMP

Un paquet HNMP comprend le champ CCP payload field (charge utile CCP). Lorsque le programme d'application transmet un paquet HNMP, il doit fixer à 0x01 le type de charge utile des champs de type de l'en-tête CCP pour informer que le type de charge utile CCP est le paquet HNMP.

La Figure 11 présente le format des paquets HNMP. Un paquet HNMP est constitué d'un en-tête HNMP de huit octets et de la charge utile HNMP. L'en-tête HNMP contient quatre champs: Transaction ID (Identificateur de transaction), HNMP Command (Commande HNMP), Reserved (Réservé) et HNMP Payload Length (Longueur de charge utile HNMP).



IEC 2081/07

Figure 11 – Format des paquets HNMP

#### 8.2.1 Transaction ID (Identificateur de transaction) (TID)

Lorsqu'un programme d'application envoie et reçoit plusieurs paquets de requête et de réponse, l'identificateur de transaction permet au programme d'application de trouver le paquet de réponse au paquet de requête correspondant. Le programme d'application peut

assigner des valeurs aléatoires à l'identificateur de transaction. La taille de l'identificateur de transaction est de deux octets.

### **8.2.2 HNMP command (Commande HNMP) (HNMPCMD)**

La commande HNMP fournit des fonctions concernant l'enregistrement d'un serveur domestique, l'enregistrement d'un dispositif, la gestion d'un dispositif, la vérification d'activité les informations d'adresse et les informations de trafic. La commande HNMP a une longueur d'un octet.

### **8.2.3 Champ Reserved (Réservé) (RSV)**

Le champ de réservation d'un octet est réservé à une utilisation future.

### **8.2.4 Champ HNMP payload length (Longueur de charge utile HNMP) (HNMPPLEN)**

Le champ HNMP payload length (Longueur de charge utile HNMP) de quatre octets indique la taille en octets de la charge utile HNMP. La longueur de la charge utile HNMP est un champ de quatre octets.

### **8.2.5 HNMP payload (Charge utile HNMP)**

Une commande HNMP peut ou non nécessiter une charge utile HNMP selon le type de commande HNMP.

DEV\_REG\_REQ, DEV\_REG\_RES, ADD\_DEV, DEL\_DEV et DEV\_INFO\_RES sont des exemples de paquets nécessitant une charge utile HNMP.

## **8.3 Enregistrement d'un serveur domestique**

Le HNMP fournit la fonction d'enregistrement de serveur domestique. L'enregistrement d'un serveur domestique se réfère au processus de détermination des adresses de domaine parmi les serveurs domestiques en utilisant les paquets de signalisation des serveurs domestiques constituant des serveurs de domaines multiples sous l'environnement de réseau à plusieurs serveurs domestiques (HS).

La présente norme envisage uniquement l'environnement de réseau à plusieurs serveurs domestiques (HS) et ne contient pas d'informations détaillées concernant l'enregistrement d'un serveur domestique.

## **8.4 Enregistrement d'un dispositif**

L'enregistrement d'un dispositif se réfère au processus de mémorisation par un serveur domestique d'un nom de dispositif lisible par un être humain et d'allocation d'un identificateur de dispositif utilisé dans la configuration d'adressage de CCP à un dispositif de CCP. Lorsqu'un nouveau dispositif de CCP doit être lié à un réseau domestique, il doit passer par le processus d'enregistrement de dispositif. Ceci permet au dispositif de CCP d'avoir tous les champs d'adresse CCP tels que Domain Address (Adresse de domaine), Cluster Address (Adresse de grappe) et Device ID (Identificateur de dispositif), par l'intermédiaire desquels le serveur domestique est capable d'avertir les autres dispositifs de CCP qu'un nouveau dispositif de CCP a été connecté au réseau domestique.

Pour exécuter le processus d'enregistrement de dispositif, on doit allouer une adresse unique à tous les dispositifs de CCP du réseau domestique. En conséquence, lors de l'exécution du processus d'enregistrement de dispositif, le serveur domestique doit être capable d'allouer différents identificateurs de dispositif à des dispositifs de CCP avec des adresses de domaine et de grappe identiques.

Les commandes associées à l'enregistrement de dispositif sont les paquets Device Registration Request (Requête d'enregistrement de dispositif) (DEV\_REG\_REQ) et Device Registration Response (Réponse à une requête d'enregistrement de dispositif) (DEV\_REG\_RES).

La Figure 12 représente le format des paquets et la Figure 13 est un exemple de séquence de commandes concernant l'enregistrement de dispositif.

#### 8.4.1 Paquet Device registration request (Requête d'enregistrement de dispositif) (DEV\_REG\_REQ)

Chaque dispositif de CCP à enregistrer dans un réseau domestique doit diffuser un paquet DEV\_REG\_REQ à son réseau en grappe. Le paquet diffusé est reçu par l'interface de serveur domestique (HSI) du serveur domestique et délivré à la couche CCP. À ce moment, le dispositif de CCP doit remplir la charge utile d'un paquet DEV\_REG\_REQ avec son nom sous forme de texte, lisible par un être humain, la taille de l'adresse physique (ou adresse de réseau) et l'adresse physique (ou adresse de réseau). Comme représenté à la Figure 12, un champ support protocol (protocole de prise en charge) est utilisé pour indiquer les protocoles que le dispositif de CCP prend également en charge parmi UHCP, HDSP et HMSP. Si le CCP prend en charge UHCP et HDSP, il convient alors que les champs UHCP et HDSP soient mis à 1.

L'interface de serveur domestique (HSI) doit communiquer Domain Address (Adresse de domaine), Cluster Address (Adresse de grappe), un Device ID (Identificateur de dispositif) nouvellement alloué et son adresse physique (ou adresse de réseau) au dispositif de CCP demandant l'enregistrement. L'interface de serveur domestique (HSI) doit également enregistrer dans la table d'adresses l'adresse CCP et l'adresse physique (ou adresse de réseau) du dispositif de CCP, pour la communication mutuelle entre le dispositif de CCP et l'interface de serveur domestique (HSI) lorsque le processus d'enregistrement est terminé.

La raison pour laquelle un dispositif et une interface de serveur domestique (HSI) échangent leurs adresses physiques (ou adresses de réseau) est qu'ils utilisent tous deux des adresses physiques ou des adresses de réseau lorsqu'ils communiquent entre eux.

Dans le cas où les couches de protocole inférieures d'un dispositif ont à la fois une adresse de réseau et une adresse physique, il est facile d'utiliser l'adresse de réseau plutôt que l'adresse physique lors de la mise en œuvre d'une interface entre la couche CCP et les couches de protocole inférieures. Par ailleurs, dans le cas où les couches de protocole inférieures d'un dispositif ne possèdent qu'une adresse physique, il n'existe aucun autre moyen que d'utiliser l'adresse physique lors de l'interfaçage entre la couche CCP et les couches de protocole inférieures.

La taille de l'adresse physique ou de l'adresse de réseau est exprimée en octets. Le champ HNMP command (Commande HNMP) d'un paquet DEV\_REG\_REQ est égal à 0x31.

S'il n'y a pas de réponse pendant une certaine période de temporisation après envoi d'un paquet DEV\_REG\_REQ, le dispositif de CCP à enregistrer effectue des relances. S'il n'y a toujours pas de réponse, on considère que le serveur domestique n'est pas prêt et la procédure est répétée après une période d'attente.

En ce qui concerne la période de temporisation, le nombre de relances et les périodes d'attente ne sont pas spécifiés en détail dans la présente norme car ils concernent la vitesse de communication influencée par la puissance de traitement d'un dispositif et la vitesse du réseau physique.

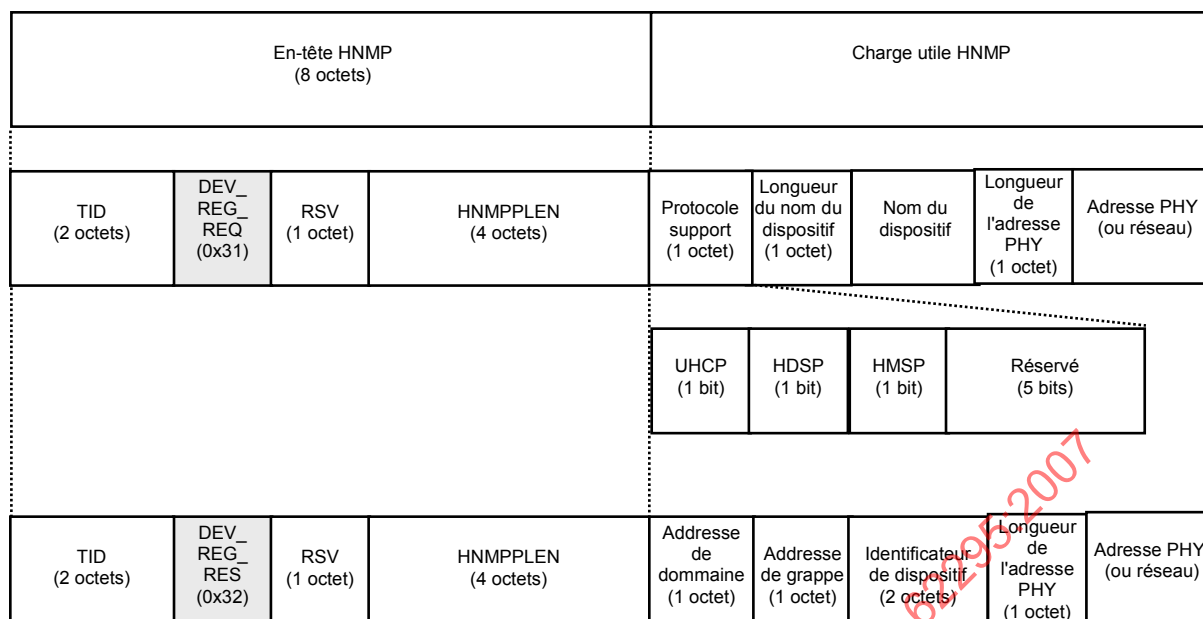


Figure 12 – Paquets DEV\_REG\_REQ et DEV\_REG\_RES

IEC 2082/07

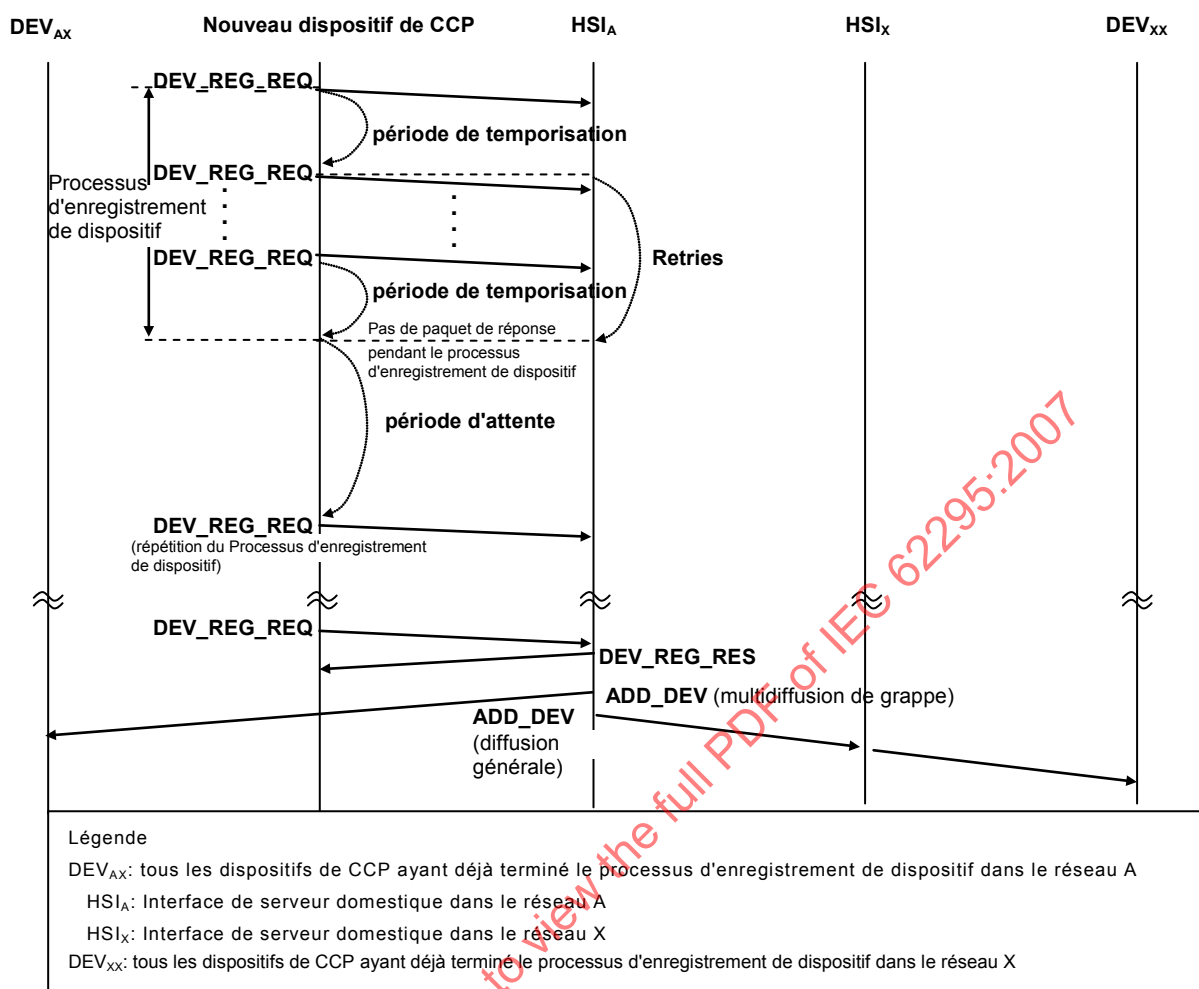
#### 8.4.2 Paquet Device registration response (Réponse à une requête d'enregistrement de dispositif) (DEV\_REG\_RES)

Lors de la réception d'un paquet DEV\_REG\_REQ, l'interface de serveur domestique (HSI) doit renvoyer en réponse pendant la période de temporisation un paquet DEV\_REG\_RES au dispositif émetteur de DEV\_REG\_REQ. L'interface de serveur domestique (HSI) doit être capable de mémoriser l'adresse physique (ou adresse de réseau) du dispositif, incluse dans la charge utile du paquet DEV\_REG\_REQ en la mappant avec adresse CCP.

De plus, l'interface de serveur domestique (HSI) doit écrire dans la charge utile HNMP ses propres adresse de domaine, adresses de grappe, identificateur de dispositif à allouer au dispositif, taille de l'adresse physique (ou adresse de réseau) et adresse physique (ou adresse de réseau). La taille de l'adresse physique (ou adresse de réseau) est indiquée en octets. Pour envoyer physiquement le paquet DEV\_REG\_RES, l'interface de serveur domestique (HSI) doit utiliser l'adresse physique (ou adresse de réseau) du dispositif de CCP comme adresse physique (ou adresse de réseau) de destination.

Après avoir reçu le paquet DEV\_REG\_RES, le dispositif de CCP configure ses propres adresse de domaine, adresse de grappe et identificateur de dispositif et mémorise l'adresse physique (ou adresse de réseau) de l'interface de serveur domestique (HSI) responsable de son réseau en grappe. Désormais, lorsque le dispositif de CCP transmet des données à un autre dispositif de CCP dans un réseau hétérogène, il doit simplement configurer l'adresse physique (ou adresse de réseau) de l'interface de serveur domestique (HSI) comme adresse physique (ou adresse de réseau) de destination au niveau de la couche physique (ou réseau). L'interface de serveur domestique (HSI) peut ensuite recevoir les données au niveau de la couche physique et extrait un paquet CCP au niveau de la couche CCP. Enfin, l'interface de serveur domestique (HSI) peut acheminer le paquet à une autre HSI à laquelle est lié le dispositif de CCP.

Le champ HNMP command (Commande HNMP) du paquet DEV\_REG\_RES est égal à 0x32.



IEC 2083/07

**Figure 13 – Exemple de séquence de commandes HNMP pour enregistrement de dispositif**

## 8.5 Gestion de dispositif

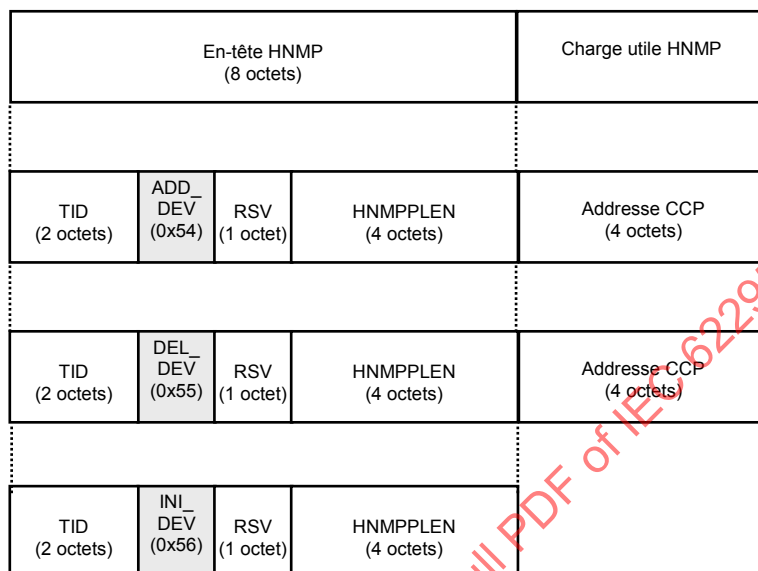
Le HNMP fournit aux interfaces de serveur domestique (HSI) et aux dispositifs de CCP la fonction de gestion de table d'adresses, à l'aide de commandes permettant d'ajouter et de supprimer un dispositif. Il peut également initialiser un dispositif de CCP particulier par l'intermédiaire de la commande Device Initialization (Initialisation de dispositif).

Les exigences concernant les interfaces de serveur domestique (HSI) (ou dispositifs de CCP) recevant des paquets Add (ou Delete) Device (Ajouter ou supprimer dispositif) sont les suivantes:

- L'interface de serveur domestique (HSI) doit être capable de délivrer le paquet Add (ou Delete) Device (Ajouter ou supprimer dispositif) à une autre HSI du réseau en grappe.
- L'interface de serveur domestique (HSI) doit être capable de délivrer le paquet Add (ou Delete) Device (Ajouter ou supprimer dispositif) d'une autre HSI à tous les dispositifs situés dans son propre réseau en grappe.
- L'interface de serveur domestique (HSI) (ou le dispositif de CCP) doit être capable d'ajouter une nouvelle adresse CCP à la table d'adresses qu'il ou elle gère.
- L'interface de serveur domestique (HSI) (ou le dispositif de CCP) recevant un paquet Delete Device (Supprimer dispositif) doit être capable de supprimer l'adresse du dispositif de CCP correspondant dans la table d'adresses.

- Les fonctions concernant l'ajout et la suppression d'un dispositif ne doivent être appliquées qu'à un nombre limité d'interfaces de serveur domestique (HSI) et de dispositifs de CCP avec des capacités de traitement et une mémoire suffisantes.

Le paquet HNMP concernant l'ajout, la suppression et l'initialisation de dispositif est constitué des paquets Add Device (Ajouter dispositif) (ADD\_DEV), Delete Device (Supprimer dispositif) (DEL\_DEV) et Initialize Device (Initialiser dispositif) (INI\_DEV) comme indiqué à la Figure 14.



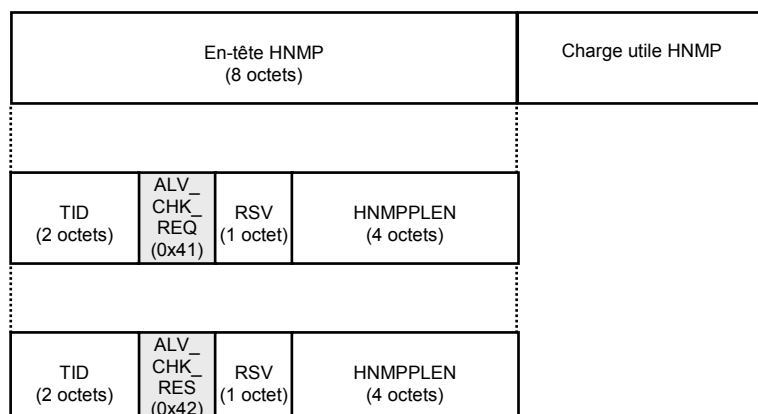
IEC 2084/07

**Figure 14 – Paquets ADD\_DEV, DEL\_DEV et INI\_DEV**

Un dispositif de CCP (ou une interface de serveur domestique (HSI)) peut évaluer l'état de fonctionnement propre du dispositif de CCP particulier et l'état d'une connexion au réseau domestique en utilisant la commande Alive-Check (Vérification d'activité) du HNMP.

Le dispositif de CCP recevant une commande Alive-Check (Vérification d'activité) doit être capable d'informer qu'il est toujours connecté au réseau domestique et qu'il fonctionne correctement. De plus, pour fournir les fonctions Plug-and-Play (Prêt à l'emploi) (PnP) des dispositifs de CCP, l'interface de serveur domestique (HSI) doit contrôler régulièrement le bon fonctionnement des dispositifs de CCP dans son réseau en grappe en utilisant la commande Alive-Check (Vérification d'activité).

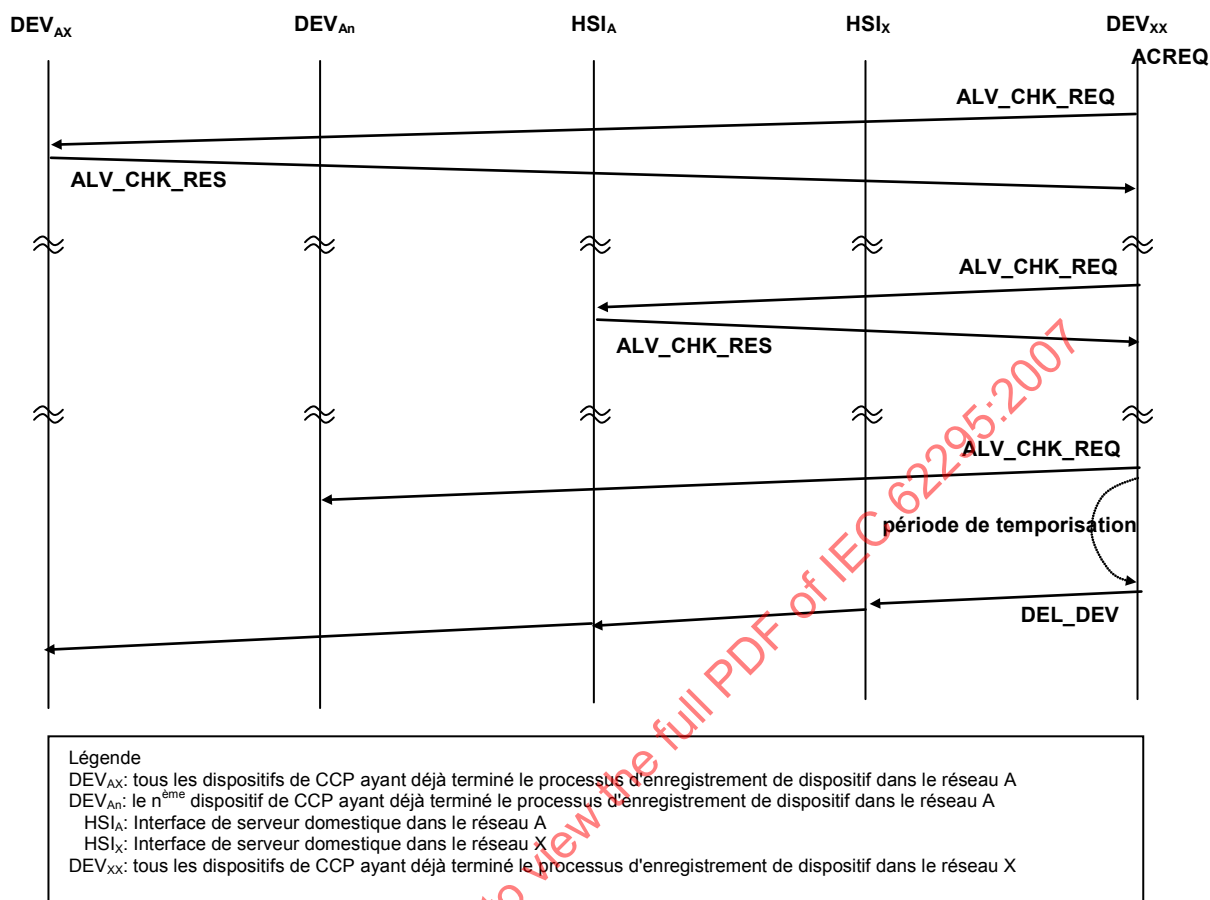
Les paquets HNMP relatifs à la commande Alive-Check (Vérification d'activité) sont Alive-Check Request (Requête de vérification d'activité) (ALV\_CHK\_REQ) et Alive-Check Response (Réponse à une requête de vérification d'activité) (ALV\_CHK\_RES), comme indiqué à la Figure 15.



IEC 2085/07

**Figure 15 – Paquets ALV\_CHK\_REQ et ALV\_CHK\_RES**

La Figure 16 montre un exemple de séquence de commandes d'un paquet HNMP concernant la gestion de dispositif.



IEC 2086/07

Figure 16 – Exemple de séquence de commandes HNMP pour la gestion de dispositif

### 8.5.1 Paquet Add Device (Ajout de dispositif) (ADD\_DEV)

Un paquet ADD\_DEV est utilisé pour informer les autres dispositifs de CCP ou interfaces de serveur domestique (HSI) qu'un nouveau dispositif de CCP est connecté au réseau domestique. Il n'est pas nécessaire que les interfaces de serveur domestique (HSI) ou les dispositifs de CCP recevant le paquet ADD\_DEV envoient un paquet de réponse.

Avant transmission, l'adresse CCP de quatre octets du dispositif de CCP nouvellement connecté au réseau domestique est fournie dans la charge utile HNMP du paquet ADD\_DEV. Le paquet ADD\_DEV est habituellement transmis avec le type de diffusion des champs de type d'en-tête CCP sous forme de diffusion générale. Un dispositif de CCP recevant le paquet ADD\_DEV peut le traiter ou le supprimer en fonction de sa capacité de gestion de table d'adresses. Les dispositifs avec une capacité de gestion de table d'adresses et un panneau d'affichage doivent être capables de mettre à jour la table d'adresses et d'indiquer sur le panneau d'affichage qu'un nouveau dispositif de CCP a été ajouté. Le champ HNMP command (Commande HNMP) du paquet ADD\_DEV est égal à « 0x54 ».

### 8.5.2 Paquet Delete device (Suppression de dispositif) (DEL\_DEV)

Un paquet DEL\_DEV est utilisé pour informer les autres dispositifs de CCP et interfaces de serveur domestique (HSI) situés dans le réseau domestique qu'un dispositif de CCP particulier a été supprimé du réseau domestique. Par exemple, un dispositif de CCP (ou une

interface de serveur domestique (HSI)) transmet un paquet DEL\_DEV s'il a déterminé qu'un dispositif de CCP particulier n'existe plus dans le réseau domestique en raison d'un dysfonctionnement ou pour d'autres raisons. Il n'est pas nécessaire que les interfaces de serveur domestique (HSI) (ou les dispositifs de CCP) recevant un paquet DEL\_DEV envoient un paquet de réponse. Avant transmission, l'adresse CCP de quatre octets du dispositif de CCP supprimé au réseau domestique est fournie dans la charge utile HNMP du paquet DEL\_DEV. Le paquet DEL\_DEV est habituellement transmis avec le type de diffusion des champs de type d'en-tête CCP sous forme de diffusion générale. Un dispositif de CCP recevant le paquet DEL\_DEV peut le traiter ou le supprimer en fonction de sa capacité de gestion de table d'adresses. Les dispositifs avec une capacité de gestion de table d'adresses et un panneau d'affichage doivent être capables de mettre à jour la table d'adresses et d'indiquer sur le panneau d'affichage qu'un dispositif de CCP particulier a été supprimé. Le champ HNMP command (Commande HNMP) du paquet DEL\_DEV est égal à « 0x55 ».

### **8.5.3 Paquet Initialize Device (Initialiser dispositif) (INI\_DEV)**

Un paquet INI\_DEV peut être envoyé d'un dispositif CCP ou d'une interface de serveur domestique (HSI) à un autre dispositif de CCP, et le dispositif de CCP recevant un paquet INI\_DEV doit lui-même s'initialiser. Il n'est pas nécessaire que le dispositif de CCP initialisé passe par un autre processus d'enregistrement de dispositif. En d'autres termes, l'adresse de domaine et l'identificateur de dispositif doivent pouvoir conserver leurs valeurs après initialisation. Le champ HNMP command (Commande HNMP) du paquet INI\_DEV est égal à « 0x56 ».

### **8.5.4 Paquet Alive-Check Request (Requête de vérification d'activité) (ALV\_CHK\_REQ)**

Un dispositif de CCP (ou serveur domestique) peut transmettre un paquet ALV\_CHK\_REQ à un autre dispositif de CCP (ou serveur domestique). Un paquet ALV\_CHK\_REQ est utilisé pour vérifier si un dispositif particulier est connecté au réseau domestique et fonctionne correctement. Un dispositif de CCP ou un serveur domestique recevant un paquet ALV\_CHK\_REQ doit renvoyer un paquet de réponse dans les deux secondes. Pour prendre en charge les fonctions Plug-and-Play (Prêt à l'emploi) (PnP) des dispositifs, une interface de serveur domestique (HSI) doit envoyer régulièrement des paquets ALV\_CHK\_REQ aux dispositifs situés dans son réseau en grappe et recevoir les réponses. Si une interface de serveur domestique (HSI) ne reçoit pas de réponse d'un dispositif de CCP pendant une certaine période de temporisation après avoir envoyé un paquet ALV\_CHK\_REQ, elle relance  $N$  fois le processus. S'il n'y a pas de réponse après  $N$  relances, le dispositif de CCP correspondant doit être considéré comme ayant été supprimé du réseau domestique et l'interface de serveur domestique (HSI) envoie un paquet DEL\_DEV aux autres HSI et à tous les dispositifs de CCP situés dans le réseau en grappe. Cette règle ne spécifie pas le nombre de relances ( $N$ ). Le champ de commande HNMP du paquet ALV\_CHK\_REQ est égal à « 0x41 ».

### **8.5.5 Paquet Alive-Check Response (Réponse à une requête de vérification d'activité) (ALV\_CHK\_RES)**

Un dispositif de CCP (ou une interface de serveur domestique (HSI)) recevant un paquet ALV\_CHK\_REQ doit renvoyer un paquet ALV\_CHK\_RES au dispositif d'où provient le paquet ALV\_CHK\_REQ durant une certaine période de temporisation pour l'avertir qu'il est connecté au réseau domestique et fonctionne correctement. Le champ HNMP command (Commande HNMP) du paquet ALV\_CHK\_RES est égal à « 0x42 ».

## **8.6 Informations d'adresse et de nom des dispositifs**

Lorsqu'un dispositif de CCP demande à une interface de serveur domestique (HSI) l'adresse CCP et les informations de nom pour tous les dispositifs de CCP liés au réseau domestique, le HNMP prend en charge la fonction de notification de l'adresse CCP et des informations de nom de tous les dispositifs de CCP. Cette fonction peut être utilisée par un dispositif de CCP nouvellement connecté pour demander les informations d'adresse CCP des dispositifs de

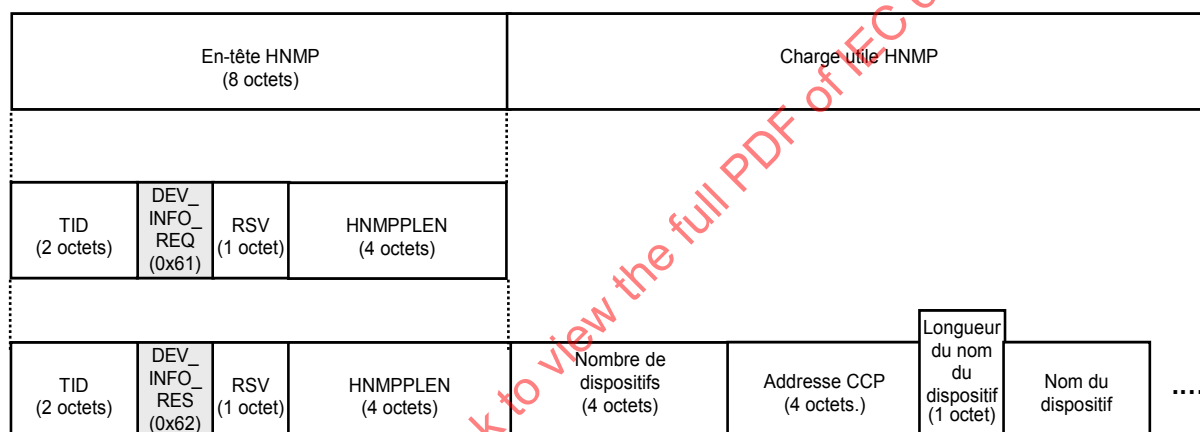
CCP existants ou par un dispositif de CCP pour mettre à jour les informations d'adresse des autres dispositifs de CCP qu'il gère.

Comme représenté à la Figure 17, il existe deux types de paquets HNMP relatifs aux informations d'adresse et de nom: Device Information Request (Requête d'informations de dispositif) (DEV\_INFO\_REQ) et Device Information Response (Réponse aux informations de dispositif) (DEV\_INFO\_RES).

Un paquet DEV\_INFO\_REQ peut être transmis d'une manière parmi deux.

Premièrement, si un dispositif de CCP particulier transmet un paquet DEV\_INFO\_REQ à une interface de serveur domestique (HSI) en monodiffusion, il peut obtenir les informations d'adresse de tous les dispositifs de CCP dans son propre réseau en grappe.

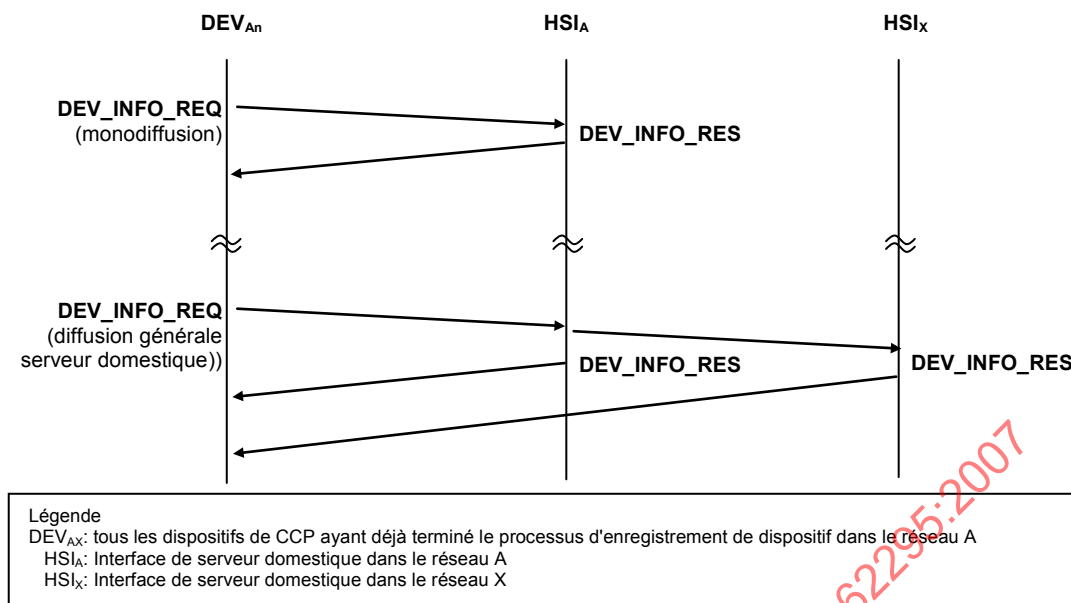
Deuxièmement, si un dispositif de CCP particulier transmet un paquet DEV\_INFO\_REQ sous forme de diffusion générale de serveur domestique (HS), il peut obtenir les informations d'adresse de tous les dispositifs de CCP dans tous les réseaux en grappe dans le réseau domestique.



IEC 2087/07

**Figure 17 – Paquets DEV\_INFO\_REQ et DEV\_INFO\_RES**

La Figure 18 montre un exemple de séquence de commandes du paquet HNMP concernant les informations d'adresse et de nom de dispositif.



IEC 2088/07

**Figure 18 – Exemple de séquence de commandes HNMP pour récupérer les informations d'adresse et de nom de dispositif**

#### 8.6.1 Paquet Device address and name information request (Requête d'informations d'adresse et de nom de dispositif) (DEV\_INFO\_REQ)

Un paquet DEV\_INFO\_REQ est utilisé par un dispositif de CCP avec capacité de gestion de table d'adresses pour demander à une interface de serveur domestique (HSI) les informations d'adresse et de nom de tous les dispositifs de CCP liés au réseau en grappe.

Une interface de serveur domestique (HSI) recevant un paquet DEV\_INFO\_REQ doit écrire dans la charge utile HNMP du paquet de réponse les informations de nom et d'adresse CCP de tous les dispositifs de CCP enregistrés dans son réseau en grappe et renvoyer le paquet de réponse au dispositif de CCP ayant envoyé le paquet de requête. Le champ HNMP command (Commande HNMP) du paquet DEV\_INFO\_REQ est égal à « 0x61 ».

#### 8.6.2 Paquet Device address and name information response (Réponse aux informations d'adresse et de nom de dispositif) (DEV\_INFO\_RES)

Une interface de serveur domestique (HSI) recevant un paquet DEV\_INFO\_REQ doit renvoyer un paquet DEV\_INFO\_RES au dispositif de CCP ayant demandé les informations de nom et d'adresse des dispositifs de CCP liés à lui-même. Sur la Figure 17, le champ The number of Devices (Nombre de dispositifs) doit avoir une longueur de quatre octets et doit contenir le nombre d'informations de dispositif de CCP énumérées dans la charge utile du paquet DEV\_INFO\_RES. Les champs CCP address (adresse CCP), length of device name (longueur de nom de dispositif) et device name (nom de dispositif) représentent les informations d'un dispositif de CCP spécifique. Le champ CCP address (adresse CCP) doit avoir une longueur de quatre octets et contenir une adresse CCP d'un dispositif de CCP spécifique. La longueur du champ device name (nom de dispositif) doit être d'un octet et il contient la longueur en octets d'un nom de dispositif. Le champ device name (nom de dispositif) doit contenir le nom d'un dispositif de CCP spécifique en format texte simple. Sa taille doit être égale à la longueur du champ device name (nom de dispositif).

Un paquet DEV\_INFO\_RES peut être transmis d'une manière parmi deux.

Premièrement, si une interface de serveur domestique (HSI) ayant reçu un paquet DEV\_INFO\_REQ a *N* dispositifs liés à elle-même, alors la HSI crée un paquet DEV\_INFO\_RES dont le champ number of devices (nombre de dispositifs) est mis à 1 et elle l'envoie *N* fois pour chaque dispositif.

Deuxièmement, une interface de serveur domestique (HSI) ayant reçu un paquet DEV\_INFO\_REQ crée un seul paquet DEV\_INFO\_RES comportant les informations d'adresse et de nom de tous les dispositifs de CCP qui sont liés à celle-ci.

Le champ HNMP command (commande HNMP) du paquet DEV\_INFO\_RES est égal à « 0x62 ».

## 8.7 Autres fonctions de gestion

Outre les fonctions spécifiées ci-dessus, de nouveaux paquets de gestion peuvent être définis et appliqués lorsqu'un réseau domestique est étendu ou que de nouvelles fonctions deviennent nécessaires.

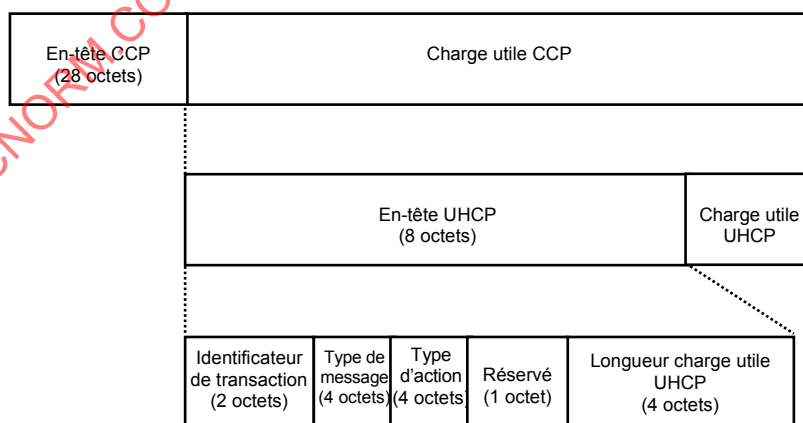
## 9 Protocole de contrôle domestique universel (UHCP)

Le CCP fournit à l'UHCP des fonctions de contrôle et de surveillance de dispositif. UHCP est un protocole à base de messages pouvant fournir des fonctions de contrôle et de surveillance des ressources d'un réseau domestique. L'un des protocoles fondamentaux fournis par CCP, UHCP est utilisé pour des services supplémentaires proposés par des dispositifs avec des capacités CCP.

Puisque la gestion des réseaux domestiques de base est effectuée par HNMP, il n'est pas nécessaire que tous les dispositifs soient incorporés dans UHCP. Si toutefois UHCP est mis en œuvre en fonction de la capacité de traitement des dispositifs de CCP, il peut fournir des services de réseau domestique supplémentaires tels que le contrôle et la surveillance.

### 9.1 Format des paquets UHCP

La Figure 19 présente le format des paquets UHCP. Le paquet UHCP est situé dans la charge utile du paquet CCP. Pour indiquer que le type de charge utile CCP est un paquet UHCP, le programme d'application UHCP doit fixer à 0x02 le type de charge utile des champs de type de l'en-tête CCP. Un paquet UHCP est constitué d'un en-tête UHCP de huit octets et de la charge utile UHCP. L'en-tête UHCP contient cinq champs: Transaction ID (identificateur de transaction), message type (type de message), action type (type d'action), reserved (réservé) et UHCP payload length (longueur de charge utile UHCP).



IEC 2089/07

Figure 19 – Format des paquets UHCP

#### 9.1.1 Transaction ID (identificateur de transaction) (TID)

Lorsqu'un programme d'application envoie et reçoit plusieurs paquets de requête et de réponse, l'identificateur de transaction est utilisé par le programme d'application pour trouver

un paquet de réponse à un paquet de requête particulier. Le programme d'application peut assigner des valeurs aléatoires à l'identificateur de transaction. L'identificateur de transaction est un champ de deux octets.

### 9.1.2 Message type (type de message) (MT) et action type (type d'action) (AT)

Le champ message type (type de message) a une longueur de quatre bits et il est constitué du message execution (exécution), du message query (demande) et du message notification (notification). Le champ action type (type d'action) a une longueur de quatre bits et il indique les actions détaillées en fonction du type de message.

La Figure 20 représente le type de message et le type d'action définis par l'UHCP.

| UHCP<br>(8 octets) |               |                       |     |          | Charge utile<br>UHCP |
|--------------------|---------------|-----------------------|-----|----------|----------------------|
| TID                | EXE<br>(0x1)  | REG<br>(0x1)          | RSV | UHCPPLEN | Charge utile<br>UHCP |
| TID                | EXE<br>(0x1)  | CTRL<br>(0x2)         | RSV | UHCPPLEN | Charge utile<br>UHCP |
| TID                | EXE<br>(0x1)  | RES<br>OK<br>(0xE)    | RSV | UHCPPLEN |                      |
| TID                | EXE<br>(0x1)  | RES<br>NOK<br>(0xF)   | RSV | UHCPPLEN |                      |
| TID                | QUE<br>(0x2)  | REG<br>STAT<br>(0x1)  | RSV | UHCPPLEN |                      |
| TID                | QUE<br>(0x2)  | CTRL<br>STAT<br>(0x2) | RSV | UHCPPLEN |                      |
| TID                | QUE<br>(0x2)  | ALL<br>STAT<br>(0x3)  | RSV | UHCPPLEN |                      |
| TID                | QUE<br>(0x2)  | RES<br>OK<br>(0xE)    | RSV | UHCPPLEN | Charge utile<br>UHCP |
| TID                | QUE<br>(0x2)  | RES<br>NOK<br>(0xF)   | RSV | UHCPPLEN |                      |
| TID                | NTFY<br>(0x3) | N/A<br>(0x0)          | RSV | UHCPPLEN | Charge utile<br>UHCP |

IEC 2090/07

**Figure 20 – Champs message type (type de message) et action type (type d'action) d'un paquet UHCP**

### 9.1.3 Champ reserved (réservé) (RSV)

Le champ reserved (réservé) d'un octet est réservé à une utilisation future.

### 9.1.4 Champ UHCP payload length (longueur de charge utile UHCP) (UHCPPLEN)

Le champ UHCP payload length (longueur de charge utile UHCP) de quatre octets indique la taille en octets de la charge utile UHCP. La taille du champ longueur est de quatre octets.

### 9.1.5 UHCP payload (charge utile UHCP)

UHCP payload (charge utile UHCP) peut être requis ou non selon le type de message et le type d'action.

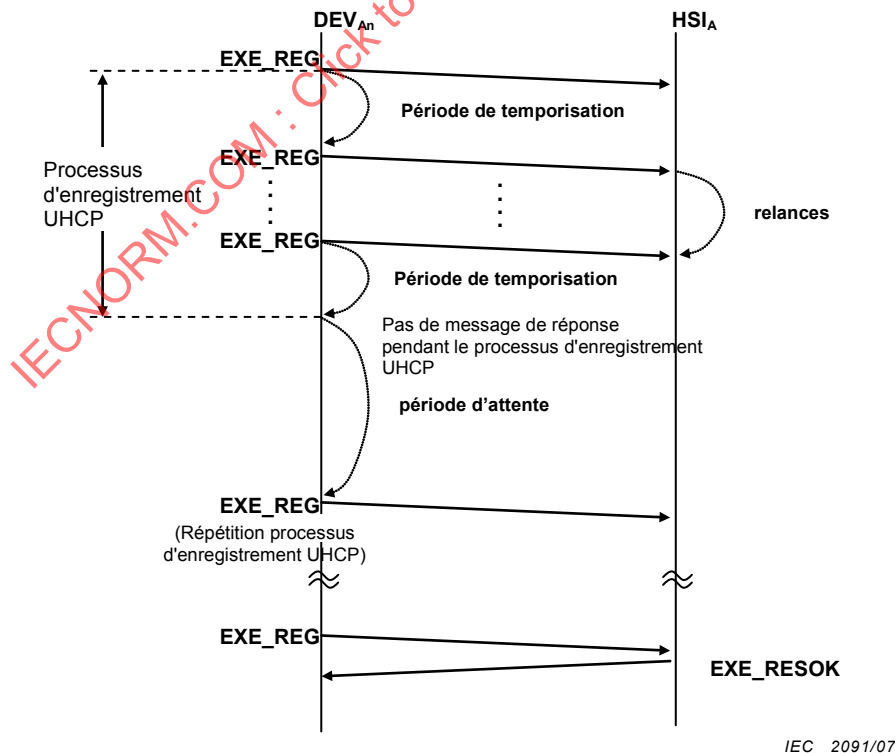
## 9.2 Messages execution (exécution) (EXE)

Si le champ MT de l'en-tête UHCP est égal à 0x1, il indique que le contenu de la charge utile UHCP est un message exécution. Il existe quatre types d'actions selon le message exécution: EXE\_REG, EXE\_CTRL, EXE\_RESOK et EXE\_RESNOK.

### 9.2.1 Exécution de l'enregistrement (EXE\_REG)

Le champ AT d'un message EXE\_REG est égal à 0x1. Pour qu'un dispositif prenant en charge UHCP utilise les fonctions UHCP après avoir terminé le processus d'enregistrement du dispositif HNMP, il doit d'abord enregistrer ses éléments de contrôle et attributs de dispositif tels que le type de dispositif, l'emplacement, le fabricant et la technologie de réseau, dans l'interface de serveur domestique (HSI) dans le réseau de domaine. Le message EXE\_REG est utilisé pour enregistrer ce type d'informations. Les attributs de dispositif et les éléments de contrôle sont indiqués dans la charge utile.

Comme représenté à la Figure 21, une interface de serveur domestique (HSI) recevant un message EXE\_REG doit envoyer un message de réponse durant la période de temporisation. Si le dispositif ayant envoyé le message EXE\_REG ne reçoit pas de réponse durant la période de temporisation, il effectue des relances. S'il ne parvient pas à recevoir de réponse après les relances d'EXE\_REG, le serveur domestique est considéré comme problématique et le processus est répété au bout de trente secondes. Puisque le dispositif doit enregistrer ses attributs par l'intermédiaire du processus afin d'exécuter les fonctions UHCP, s'il n'y a pas de réponse de l'interface de serveur domestique (HSI), il continue à répéter le processus jusqu'à ce que l'enregistrement soit terminé.



IEC 2091/07

Figure 21 – Exemple de processus d'enregistrement

### 9.2.2 Exécution du contrôle (EXE\_CTRL)

Le champ AT d'un message EXE\_CTRL est égal à 0x2. Un dispositif recevant un paquet EXE\_CTRL doit exécuter la commande en fonction du contenu de la charge utile et envoyer un message de réponse correspondant.

La Figure 22 représente la séquence d'un message EXE\_CTRL. Si le dispositif de CCP ayant envoyé un message EXE\_CTRL ne reçoit pas de message de réponse EXE\_RESOK durant la période de temporisation, il envoie un message EXE\_RESNOK au programme d'application UHCP. Même si le programme d'application UHCP ne reçoit pas de paquet de réponses à un message EXE\_CTRL, il n'effectue pas de relance. Toutefois, un programme d'application utilisant les fonctions UHCP peut décider de la période de temporisation et effectuer une relance en fonction d'une requête de l'utilisateur.

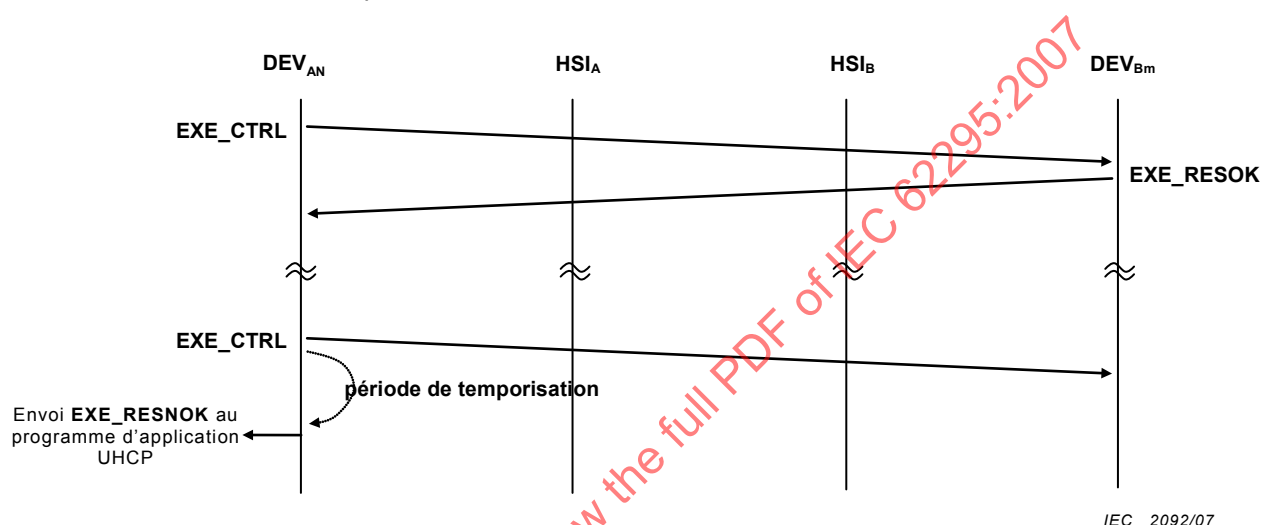


Figure 22 – Exemple de message EXE\_CTRL

### 9.2.3 Response OK (Réponse correcte) (EXE\_RESOK)

Le champ AT d'un message EXE\_RESOK est égal à 0xE. Un dispositif recevant un paquet EXE\_REG ou EXE\_CTRL doit terminer l'enregistrement de dispositif ou exécuter une commande et renvoyer un message EXE\_RESOK indiquant que la tâche a été exécutée avec succès. Un message EXE\_RESOK ne comporte pas de charge utile UHCP. (Se référer à la Figure 20)

### 9.2.4 Response NOK (Réponse incorrecte) (EXE\_RESNOK)

Le champ AT d'un message EXE\_RESNOK est égal à 0xF. Un dispositif recevant un paquet EXE\_REG ou EXE\_CTRL doit renvoyer un message EXE\_RESNOK pour indiquer que le processus d'enregistrement ou une commande a échoué. Dans le cas du message EXE\_REG, si le dispositif ayant envoyé un message EXE\_REG ne parvient pas à recevoir de réponse EXE\_RESOK ou EXE\_RESNOK en raison d'un serveur domestique instable, le dispositif doit être capable d'envoyer de lui-même un message EXE\_RESNOK à l'application UHCP. Un message EXE\_RESNOK ne comporte pas de charge utile UHCP, comme indiqué à la Figure 20.

## 9.3 Messages query (demande) (QUE)

Si le champ MT d'un en-tête UHCP est égal à 0x2, il indique que le contenu de la charge utile UHCP est un message de demande ou de surveillance. Il existe cinq types de messages d'exécution dans le type d'action, en fonction du message de demande: QUE\_REGSTAT, QUE\_CTRLSTAT, QUE\_ALL, QUE\_RESOK et QUE\_RESNOK, comme décrit ci-dessous.

### 9.3.1 Query of registration status (demande d'état d'enregistrement) (QUE\_REGSTAT)

Le champ AT d'un message QUE\_REGSTAT est égal à 0x1. Il est utilisé par un serveur domestique ou par un autre dispositif pour demander les attributs d'un dispositif particulier. Un dispositif recevant un message QUE\_REGSTAT doit fournir ses informations d'attributs dans la charge utile UHCP du message QUE\_RESOK et le renvoyer. S'il n'y a pas de réponse pendant la période de temporisation, le dispositif ayant envoyé le message QUE\_REGSTAT doit envoyer un message QUE\_RESNOK au programme d'application UHCP. La Figure 23 montre un exemple de message QUE\_REGSTAT. Un message QUE\_REGSTAT ne comporte pas de charge utile UHCP, comme indiqué à la Figure 20.

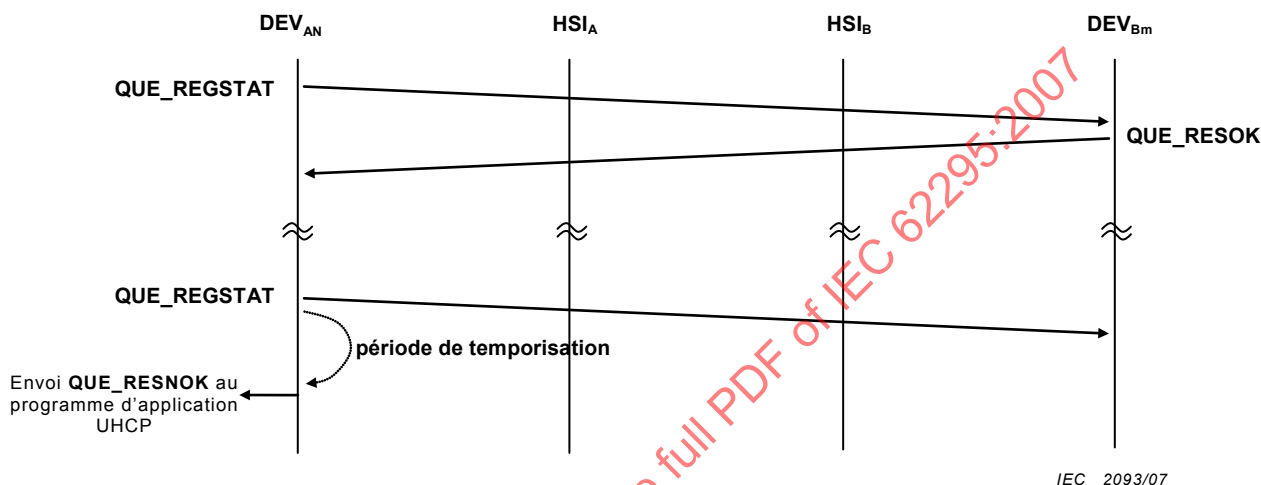


Figure 23 – Exemple de message QUE\_REGSTAT

### 9.3.2 Query of control status (demande d'état de contrôle) (QUE\_CTRLSTAT)

Le champ AT d'un message QUE\_CTRLSTAT est égal à 0x2. Il est utilisé par un serveur domestique ou par un autre dispositif pour demander des informations concernant l'état d'une commande d'un dispositif particulier. Un dispositif recevant un message QUE\_CTRLSTAT doit fournir ses informations d'état de commande dans la charge utile UHCP du message QUE\_RESOK et la renvoyer. S'il n'y a pas de réponse pendant la période de temporisation, le dispositif ayant envoyé le message QUE\_CTRLSTAT doit envoyer un message QUE\_RESNOK au programme d'application. La Figure 24 montre un exemple de message QUE\_CTRLSTAT. Un message QUE\_CTRLSTAT ne comporte pas de charge utile UHCP, comme indiqué à la Figure 20.

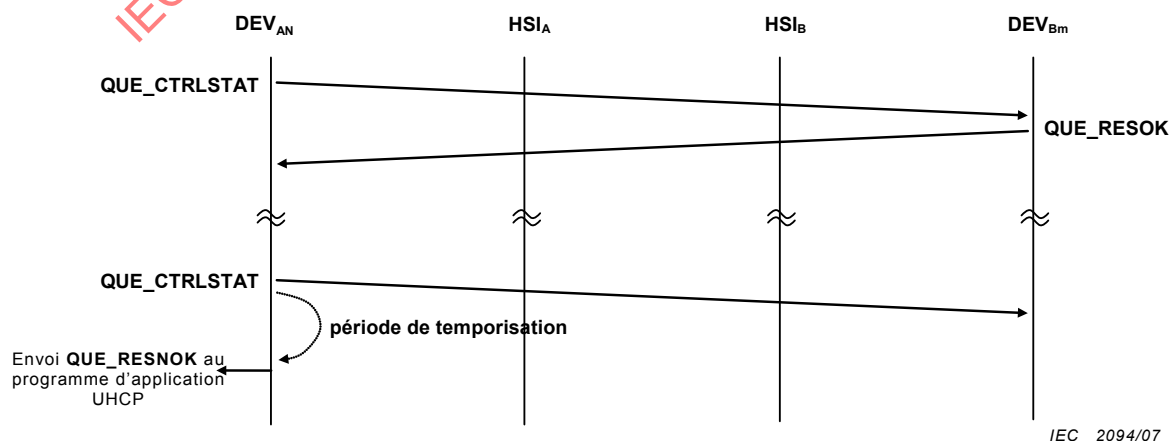


Figure 24 – Exemple de message QUE\_CTRLSTAT