INTERNATIONAL STANDARD

ISO 24102-4

First edition
2013-07-01

AMENDMENT 1
2017-04

Intelligent transport systems — Communications access for land mobiles (CALM) — ITS station management —

Part 4: Station-internal management communications

AMENDMENT 1

Systèmes intelligents de transport — Accès aux communicat ions des services mobiles terrestres (CALM) — Gestion des stations ITS —

Partie 4: Communications de gestion interne à la station

AMENDEMENT 1

Reference number
ISO 24102-4:2013/Amd.1:2017(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

Amendment 1 to ISO 24102-4:2013 was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems*.

A list of all parts in the ISO 24102 series can be found on the ISO website.

Intelligent transport systems — Communications access for land mobiles (CALM) — ITS station management —

Part 4:
Station-internal management communications

AMENDMENT 1

Page 4, Figure 3

Replace Figure 3 by the following.

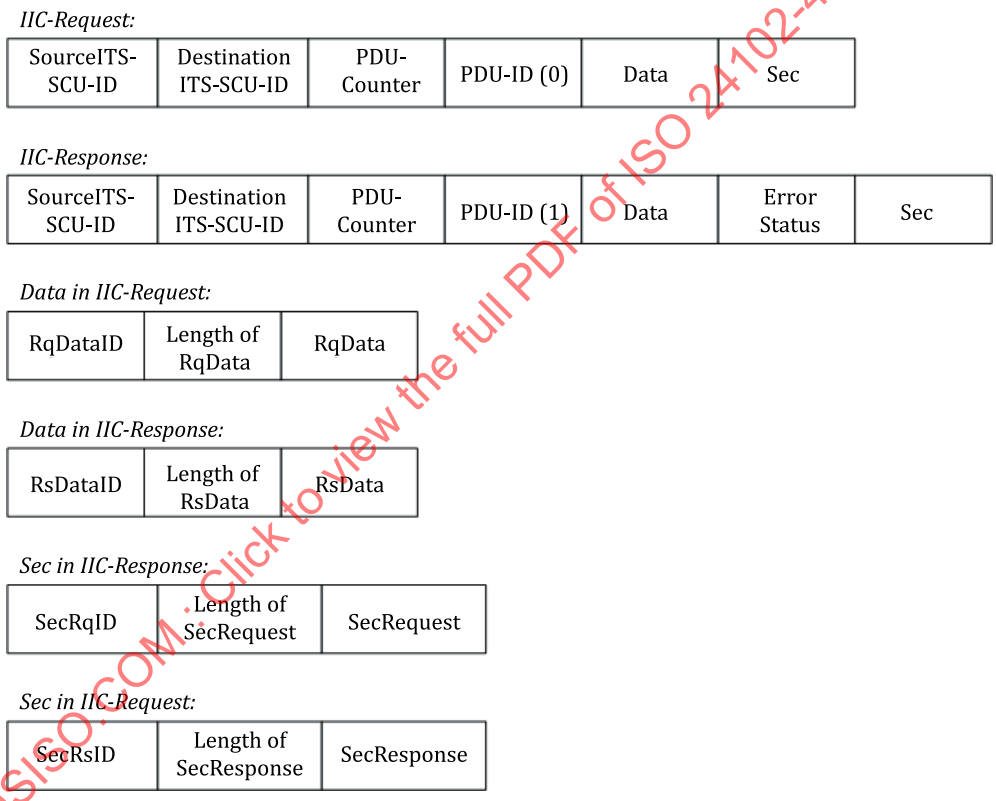

Figure 3 — IIC PDU structure

Page 4, Table 1

Replace Table 1 by the following.

PDU element	IIC-Request	IIC-Response
SourceITS-SCU-ID	ITS-SCU-ID of source ITS-SCU, which produces the request. See parameter "ITS-scuId" specified in ISO 24102-1.	ITS-SCU-ID of ITS-SCU, which produces the response.
DestinationITS-SCU-ID	ITS-SCU-ID of destination ITS-SCU, which shall evaluate the request.	Same as SourceITS-SCU-ID of related request if not requested otherwise in this document.
PDU-Counter	Even number generated from a cyclic counter at the ITS-SCU, which produces the request.	PDU-Counter of related request incremented by one.
PDU-ID	One octet INTEGER value 0.	One octet INTEGER value 1.
Data	Data type identifier RqDataID followed by length of RqData in Integer multiples of an octet and RqData.	Data type identifier RsDataID followed by length of RsData in Integer multiples of an octet and RsData.
ErrorStatus	Not existent.	Existent! 0: No error happened > 0: Number indicating type of error.
Seq	Security type identifier SecRqID followed by the length of SecRequest in Integer multiples of an octet and SecRequest.	Security type identifier SecRsID followed by the length of SecResponse in Integer multiples of an octet and SecResponse.

Page 5, Table 2

Replace Table 2 by the following.

SourceITS-SCU-ID	Destination-ITS-SCU-ID	Description
0		Reserved. Used to indicate "own/local ITS-SCU". Must not be used in communications with other ITS-SCUs.
n.a.	1	ITS-SCU-ID identifying ITS-SCUs with ITS-S host role.
n.a.	2	ITS-SCU-ID identifying ITS-SCUs with ITS-S router role.
3	3	ITS test system (using IICP for upper tester access).
4 - 7		ITS-SCU-ID identifying ITS-SCUs with an implementation specific role.
8 .. 65534		ITS-SCU-ID identifying uniquely a specific ITS-SCU in an ITS station.
n.a.	65535	ITS-SCU-ID identifying all ITS-SCUs.

Page 10, Annex A

Replace Annex A with the following:

Annex A
(normative)

ASN.1 module

A.1 Overview

The following ASN.1 module is specified in this annex:

— CALMiitsscu { ISO (1) standard (0) calm-management (24102) iitsscu (4) asnm-1 (1)}

A.2 Module CALMiitsscu

This module specifies ASN.1 type definitions together with useful ASN.1 value definitions.

Unaligned packed encoding rules (PER) as specified in ISO/IEC 8825-2 shall be applied for this ASN.1 module.

In order to achieve octet alignment enabling cheap implementations, "fill" bits were defined. All fill bits shall be set to the value '0'b.

```
CALMiitsscu { iso (1) standard (0) calm-management (24102) iitsscu (4) asnm-1 (1)}
DEFINITIONS AUTOMATIC TAGS::=BEGIN

IMPORTS
CIaClass, CIclass, CIstatus, Directivity, Link-ID, LLserviceAddr, MACaddress, MedType FROM
CALMllsap {iso(1) standard(0) calm-ll-sap(21218) asnm-1 (1)}

ITS-scuId, Param24102No, Param24102 FROM CALMmanagement { iso (1) standard (0)
calm-management (24102) local (1) asnm-1 (1)}

SimFUTcmd, SimFUTreq, SimFLTcmd, SimNUTcmd, SimNLTcmd, SimIUTcmd, SimFLTreq, SimNUTreq,
SimNLTreq, SimIUTreq, ErrStatus, MF-Command-request, MF-Request-request, MN-Command-
request, MN-Request-request, MI-Command-request, MI-Request-request, MA-Command-request,
MA-Request-request, MS-Command-request, MS-Request-request, MI-Get-request, MI-Set-
request, MF-Command-confirm, MF-Request-confirm, MN-Command-confirm, MN-Request-confirm,
MI-Command-confirm, MI-Request-confirm, MA-Command-confirm, MA-Request-confirm, MS-Command-
confirm, MS-Request-confirm, MI-Get-confirm, MI-Set-confirm FROM CALMmsap {iso (1) standard
(0) calm-management (24102) msap (3) asnm-1 (1)}

-- Details on SecRq and SecRs to be provided by a standard on security
;
-- End of IMPORTS

-- Types

-- PDUs --
-- IIC-Request PDU
IIC-Request::=SEQUENCE{
       sourceITS-scuId            ITS-scuId,
       destinationITS-scuId       ITS-scuId,
       pduCounter                 PduCounter,
       requestID                  RequestID, -- value zero
       pduRequest                 PduRequest,
       secRq                      SecRq
       }

RequestID::=INTEGER{
       request       (0)
       } (0..255)


SECIICP::=CLASS{
       &ref                INTEGER(0..255), -- security type identifier
```

```
       &SecIICP
       }

SecIICPs SECIICP::={noSecurity | octString, ...}


noSecurity                  SECIICP::={&ref 0, &SecIICP NULL}
octString                   SECIICP::={&ref 1, &SecIICP OCTET STRING (SIZE(0..65535))}

SecRq::=SEQUENCE{
       secRqRef             SECIICP.&ref({SecIICPs}),
       secRq                SECIICP.&SecIICP({SecIICPs}{@secRqRef})
       }

IICPREQ::=CLASS{
       &ref                 INTEGER(0..255), -- data type identifier
       &IICPrequest
       }

IICPrequests IICPREQ::={aliveRq | mfRcmdRq | mfRreqRq | mnRcmdRq | mnRreqRq | miRcmdRq |
miRreqRq | miRgetRq | miRsetRq | mCmdRq | maRcmdRq | maRreqRq | msRcmdRq | msRreqRq |
simFutCmdRq | simFutReqRq | simNutCmdRq | simNutReqRq | simIutCmdRq | simIutReqRq |
simFltCmdRq | simFltReqRq | simNltCmdRq | simNltReqRq, ...}

PduRequest::=SEQUENCE{
       reqRef               IICPREQ.&ref({IICPrequests}), -- data type identifier
       iicpReq              IICPREQ.&IICPrequest({IICPrequests}{@reqRef})
       }

aliveRq             IICPREQ::={&ref 0, &IICPrequest ITS-SCUalive}
mfRcmdRq            IICPREQ::={&ref 1, &IICPrequest MF-Command-request}
mfRreqRq            IICPREQ::={&ref 2, &IICPrequest MF-Request-request}
mnRcmdRq            IICPREQ::={&ref 3, &IICPrequest MN-Command-request}
mnRreqRq            IICPREQ::={&ref 4, &IICPrequest MN-Request-request}
miRcmdRq            IICPREQ::={&ref 5, &IICPrequest MI-Command-request}
miRreqRq            IICPREQ::={&ref 6, &IICPrequest MI-Request-request}
miRgetRq            IICPREQ::={&ref 7, &IICPrequest MI-Get-request}
miRsetRq            IICPREQ::={&ref 8, &IICPrequest MI-Set-request}
mCmdRq              IICPREQ::={&ref 9, &IICPrequest McmdRq}
maRcmdRq            IICPREQ::={&ref 10, &IICPrequest MA-Command-request}
maRreqRq            IICPREQ::={&ref 11, &IICPrequest MA-Request-request}
msRcmdRq            IICPREQ::={&ref 12, &IICPrequest MS-Command-request}
msRreqRq            IICPREQ::={&ref 13, &IICPrequest MS-Request-request}
-- simAutCmdRq              IICPREQ::={&ref 242, &IICPrequest SimAUTcmd}
-- simAutReqRq              IICPREQ::={&ref 243, &IICPrequest SimAUTreq}
simFutCmdRq                 IICPREQ::={&ref 244, &IICPrequest SimFUTcmd}
simFutReqRq                 IICPREQ::={&ref 245, &IICPrequest SimFUTreq}
simNutCmdRq                 IICPREQ::={&ref 246, &IICPrequest SimNUTcmd}
simNutReqRq                 IICPREQ::={&ref 247, &IICPrequest SimNUTreq}
simIutCmdRq                 IICPREQ::={&ref 248, &IICPrequest SimIUTcmd}
simIutReqRq                 IICPREQ::={&ref 249, &IICPrequest SimIUTreq}
-- simAltCmdRq              IICPREQ::={&ref 250, &IICPrequest SimALTcmd}
-- simAltReqRq              IICPREQ::={&ref 251, &IICPrequest SimALTreq}
simFltCmdRq                 IICPREQ::={&ref 252, &IICPrequest SimFLTcmd}
simFltReqRq                 IICPREQ::={&ref 253, &IICPrequest SimFLTreq}
simNltCmdRq                 IICPREQ::={&ref 254, &IICPrequest SimNLTcmd}
simNltReqRq                 IICPREQ::={&ref 255, &IICPrequest SimNLTreq}

ITS-SCUalive::=SEQUENCE{
       message      AliveMessage,
       its-scuType  ITS-SCUtype
       }

AliveMessage::= INTEGER{
       alive (0),
       delete (1),
       new (255)
       } (0..255)

ITS-SCUtype::= INTEGER{
       unknown (0),
```

```
        host (1),
        router (2),
        itsTestSystem (3),
        any (255)
        } (0..255)

-- General OBJECT CLASS for management commands

MCMD::=CLASS{
        &refMcmd      INTEGER(0..255), -- allows for 256 commands
        &Mcmd
        }


MCMDREQ::=MCMD

McmdRq::=SEQUENCE{
        mCmdRef       MCMDREQ.&refMcmd({McmdReqs}),
        mcmd          MCMDREQ.&Mcmd ({McmdReqs}{@mCmdRef})
        }

McmdReqs MCMDREQ::={pingRq | vCIinfoRq | vCIupdateRq | getParam24102Rq | setParam24102Rq |
testIicpRq | testConfigRq, ...}

pingRq              MCMDREQ::={&refMcmd 0, &Mcmd PingIICP}
vCIinfoRq           MCMDREQ::={&refMcmd 1, &Mcmd VCI-info-req}
vCIupdateRq         MCMDREQ::={&refMcmd 2, &Mcmd VCI-update-req}
getParam24102Rq     MCMDREQ::={&refMcmd 3, &Mcmd Param24102No}
setParam24102Rq     MCMDREQ::={&refMcmd 4, &Mcmd Param24102}
testIicpRq          MCMDREQ::={&refMcmd 254, &Mcmd TestIicp}
testConfigRq        MCMDREQ::={&refMcmd 255, &Mcmd TestConfigIICP}

PingIICP::=OCTET STRING (SIZE(0..255))

VCI-info-req::=SEQUENCE{
        medType       MedType,
        ciaClass      CIaClass,
        ciClass       CIclass
        }

VCI-update-req::=SEQUENCE (SIZE(0..255)) OF VCI-Info

VCI-Info::=SEQUENCE{
        linkId Link-ID,
        medType       MedType,
        ciaClass      CIaClass,
        ciClass       CIclass,
        status        CIstatus
        }

TestIicp::=SEQUENCE{
        fill          BIT STRING (SIZE(6)), -- all bits set to '0'
        action        CHOICE{
-- simulation of a trigger event in the ITS-S management entity of the SUT
-- to perform remote COMMAND access to an MX-SAP in the ITS test system
-- request to send MF-Command-request to ITS test system
        mfRcmd          MFrcmd,
-- request to send MN-Command-request to ITS test system
        mnRcmd          MNrcmd,
-- request to send MI-Command-request to ITS test system
        miRcmd          MIrcmd
-- request to send MS-Command-request to ITS test system
--      msRcmd          MSrcmd,
-- request to send MA-Command-request to ITS test system
        maRcmd          MArcmd
              }
        }

-- Upper Tester TX: remote COMMAND accesses to an MX-SAP

-- MSrcmd::=SEQUENCE{
```

```
--      destination   ITS-scuId,
--      cmd           MS-Command-request
--      }
MArcmd::=SEQUENCE{
        destination   ITS-scuId,
        cmd           MA-Command-request
        }
MFrcmd::=SEQUENCE{
        destination   ITS-scuId,
        cmd           MF-Command-request
        }
MNrcmd::=SEQUENCE{
        destination   ITS-scuId,
        cmd           MN-Command-request
        }
MIrcmd::=SEQUENCE{
        destination   ITS-scuId,
        cmd           MF-Command-request
        }

TestConfigIICP::=SEQUENCE{
        location      BIT STRING (SIZE(6)),
        lowerLayers   BIT STRING (SIZE(1)),
        testMode      BIT STRING (SIZE(1))
        }
```

-- IIC-Response PDU

```
IIC-Response::=SEQUENCE{
        sourceITS-scuId             ITS-scuId,
        destinationITS-scuId        ITS-scuId,
        pduCounter                  PduCounter,
        responseID                  ResponseID, -- value 1
        pduResponse                 PduResponse,
        errorStatus                 PduErrStatus,
        secRs                       SecRs
        }

ResponseID::=INTEGER{
        response      (1)
        } (0..255)

SecRs::=SEQUENCE{
        secRsRef            SECIICP.&ref({SecIICPs}),
        secRs               SECIICP.&SecIICP({SecIICPs}{@secRsRef})
        }


IICPRES::=CLASS{
        &ref                INTEGER(0..255),
        &IICPresponse
        }

IICPresponses IICPRES::={aliveRs | mfRcmdRs | mfRreqRs | mnRcmdRs | mnRreqRs | miRcmdRs |
miRreqRs | miRgetRs | miRgetRs | miRsetRs | mCmdRs | maRcmdRs | maRreqRs | msRcmdRs |
msRreqRs | simAutCmdRs | simAutReqRs | simFutCmdRs | simFutReqRs | simNutCmdRs |
simNutReqRs | simIutCmdRs | simIutReqRs | simAltCmdRs | simAltReqRs | simFltCmdRs |
simFltReqRs | simNltCmdRs | simNltReqRs, ...}

PduResponse::=SEQUENCE{
        resRef              IICPRES.&ref({IICPresponses}), -- data type identifier
        iicpRes             IICPRES.&IICPresponse({IICPresponses}{@resRef})
        }

aliveRs             IICPRES::={&ref 0, &IICPresponse ITS-SCUalive}
mfRcmdRs            IICPRES::={&ref 1, &IICPresponse MF-Command-confirm}
mfRreqRs            IICPRES::={&ref 2, &IICPresponse MF-Request-confirm}
mnRcmdRs            IICPRES::={&ref 3, &IICPresponse MN-Command-confirm}
mnRreqRs            IICPRES::={&ref 4, &IICPresponse MN-Request-confirm}
miRcmdRs            IICPRES::={&ref 5, &IICPresponse MI-Command-confirm}
miRreqRs            IICPRES::={&ref 6, &IICPresponse MI-Request-confirm}
```



```
miRgetRs            IICPRES::={&ref 7, &IICPresponse MI-Get-confirm}
miRsetRs            IICPRES::={&ref 8, &IICPresponse MI-Set-confirm}
mCmdRs              IICPRES::={&ref 9, &IICPresponse McmdRs}
maRcmdRs            IICPRES::={&ref 10, &IICPresponse MA-Command-confirm}
maRreqRs            IICPRES::={&ref 11, &IICPresponse MA-Request-confirm}
msRcmdRs            IICPRES::={&ref 12, &IICPresponse MS-Command-confirm}
msRreqRs            IICPRES::={&ref 13, &IICPresponse MS-Request-confirm}
simAutCmdRs         IICPRES::={&ref 242, &IICPresponse NULL}
simAutReqRs         IICPRES::={&ref 243, &IICPresponse NULL}
simFutCmdRs         IICPRES::={&ref 244, &IICPresponse NULL}
simFutReqRs         IICPRES::={&ref 245, &IICPresponse NULL}
simNutCmdRs         IICPRES::={&ref 246, &IICPresponse NULL}
simNutReqRs         IICPRES::={&ref 247, &IICPresponse NULL}
simIutCmdRs         IICPRES::={&ref 248, &IICPresponse NULL}
simIutReqRs         IICPRES::={&ref 249, &IICPresponse NULL}
simAltCmdRs         IICPRES::={&ref 250, &IICPresponse NULL}
simAltReqRs         IICPRES::={&ref 251, &IICPresponse NULL}
simFltCmdRs         IICPRES::={&ref 252, &IICPresponse NULL}
simFltReqRs         IICPRES::={&ref 253, &IICPresponse NULL}
simNltReqRs         IICPRES::={&ref 255, &IICPresponse NULL}
simNltCmdRs         IICPRES::={&ref 254, &IICPresponse NULL}

MCMDRES::=MCMD

McmdRess MCMDRES::={pingRs | vCIinfoRs | vCIupdateRs | getParam24102Rs | setParam24102Rs |
testIicpRs | testConfigRs, ...}

McmdRs::=SEQUENCE{
       mCmdRef              MCMDRES.&refMcmd({McmdRess}),
       mcmd                 MCMDRES.&Mcmd ({McmdRess}{@mCmdRef})
       }

pingRs              MCMDRES::={&refMcmd 0, &Mcmd PingIICP}
vCIinfoRs           MCMDRES::={&refMcmd 1, &Mcmd VCI-info-res}
vCIupdateRs         MCMDRES::={&refMcmd 2, &Mcmd NULL}
getParam24102Rs     MCMDRES::={&refMcmd 3, &Mcmd Param24102}
setParam24102Rs     MCMDRES::={&refMcmd 4, &Mcmd ErrStatus}
testIicpRs          MCMDRES::={&refMcmd 254, &Mcmd NULL}
testConfigRs        MCMDRES::={&refMcmd 255, &Mcmd NULL}

PduErrStatus::=INTEGER{
       success (0),
       pduUnknown (1),
       duplicateITS-scuId (2),
       invalidAliveMessage (3),
       invalidITSscuType (4),
       unspecFailure (255)
       } (0..255)

VCI-info-res::=SEQUENCE (SIZE(0..255)) OF VCI-Info

-- MF-SAP --

-- MF-COMMANDs --

IICrequestTX::=IIC-Request

IICresponseTX::=IIC-Response

-- MF-REQUESTs --

IICrequestRX::=IIC-Request

IICresponseRX::=IIC-Response


-- General types --

PduCounter::=INTEGER(0..65535)

-- Values
```

```
version INTEGER(0..255)::=2 -- version of this ASN.1 module

/*
   The ASN.1 specification has been checked for conformance to the ASN.1
   standards by OSS ASN.1 Syntax Checker, and by OSS ASN-1STEP
*/

END
```

Page 13, Annex B

Replace Annex B with the following:

Annex B
(normative)

IIC PDUs

B.1 List of PDUs

Table B.1 presents an overview of all ICC PDUs. Further details are specified in the next subclauses of this annex. The column "Response" indicates whether a response PDU is mandatory (yes) or prohibited (no). A response may contain an empty Data field. See also Table 1. The column "Registration ID" indicates the reference number of the PDU type. Further PDU types may apply as being registered without the need to be listed in this document.

NOTE A registry of IICP PDU types is under preparation at ISO.

Table B.1 — PDU-ID

Type of PDU	Response	Registration ID	Comment
ITS-SCUalive	yes	0	Used to assign, maintain and delete unique ITS-SCU-ID values in a station.
MF-rcmd	yes	1	A management command MF-COMMAND issued by the ITS station management entity of the local ITS-SCU, to be forwarded to the MF-SAP of one or several remote ITS-SCUs.
MF-rreq	yes	2	A command MF-REQUEST issued by the local facilities layer, to be forwarded to the ITS station management entity in one or several remote ITS-SCUs.
MN-rcmd	yes	3	A management command MN-COMMAND issued by the ITS station management entity of the local ITS-SCU, to be forwarded to the MN-SAP of one or several remote ITS-SCUs.
MN-rreq	yes	4	A command MN-REQUEST issued by the local networking and transport layer, to be forwarded to the ITS station management entity in one or several remote ITS-SCUs.
MI-rcmd	yes	5	A management command MI-COMMAND issued by the ITS station management entity of the local ITS-SCU, to be forwarded to the MI-SAP of a remote ITS-SCUs.
MI-rreq	yes	6	A command MI-REQUEST issued by the local access layer, to be forwarded to the ITS station management entity in one or several remote ITS-SCUs.
MI-rget	yes	7	A command MI-GETPARAM issued by the ITS station management entity, to be forwarded to the MI-SAP of a remote ITS-SCU.

Table B.1 *(continued)*

Type of PDU	Response	Registration ID	Comment
MI-rset	yes	8	A command MI-SETPARAM issued by the ITS station management entity, to be forwarded to the MI-SAP of a remote ITS-SCU.
Mcmd	—	9	Request executing of a management command in the ITS-S management entity of a remote ITS-SCU. See details in the rows below.
Mcmd.Ping	yes	9	Octet string with random data to be replied.
Mcmd.VCI-info	yes	9	Request to all ITS-SCUs containing a router, to report about existing VCIs. Information to be stored in VCI list.
Mcmd.VCI-update	no	9	Information on change of VCI information to be stored in VCI list. Broadcasted to all ITS-SCUs.
Mcmd.GET-Param24102	yes	9	Retrieves the value of a management parameter Param24102 specified in ISO 24102-1 from another ITS-SCU.
Mcmd.SET-Param24102	yes	9	Set the value of a management parameter Param24102 specified in ISO 24102-1 from another ITS-SCU.
Mcmd.TestIicp	yes	9	Used for test purposes only. Simulation of a trigger event in the ITS-S management entity of the SUT to perform remote COMMAND access to an MX-SAP in the ITS test system.
Mcmd.TestConfigIICP	yes	9	Used to program the test mode in an ITS-SCU.
—	—	10 - 241	To be assigned by the registry.
SimAUTcmd	yes	242	See ISO 24102-3.
SimAUTreq	yes	243	See ISO 24102-3.
SimFUTcmd	yes	244	See ISO 24102-3.
SimFUTreq	yes	245	See ISO 24102-3.
SimNUTcmd	yes	246	See ISO 24102-3.
SimNUTreq	yes	247	See ISO 24102-3.
SimIUTcmd	yes	248	See ISO 24102-3.
SimIUTreq	yes	249	See ISO 24102-3.
SimALTcmd	yes	250	See ISO 24102-3.
SimALTreq	yes	251	See ISO 24102-3.
SimFLTcmd	yes	252	See ISO 24102-3.
SimFLTreq	yes	253	See ISO 24102-3.
SimNLTcmd	yes	254	See ISO 24102-3.
SimNLTreq	yes	255	See ISO 24102-3.

B.2 PDU details

B.2.1 ASN.1

ASN.1 details of all PDUs shall be as presented in Annex A.

B.2.2 ITS-SCUalive

Table B.2 shows details of the “Data” element in the IIC-Request PDU.

Table B.2 — ITS-SCU-id request PDU

Name	Description
AliveMessage	Indicates type of alive message: “alive” “delete” “new”
ITS-SCUtype	Indicates role of ITS-SCU: ITS-S Host ITS-S Router ITS-S Host and ITS-S Router

This request shall always be transmitted to all ITS-SCUs.

Table B.3 shows details of the “Data” element in the IIC-Response PDU.

Table B.3 — ITS-SCU-id response PDU

Name	Description
Alive Message	Same as in related request
ITS-SCUtype	Same as in related request

With “Alive Message” = “delete” no IIC-Response PDU shall be transmitted.

B.2.3 Remote SAP access

The “Data” element in the IIC-Request PDUs shall contain the SAP service primitives.

— MF-Command-request

— MN-Command-request

— MI-Command-request

— MF-Request-request

— MN-Request-request

— MI-Request-request

— MI-Get-request

— MI-Set-request

specified in ISO 24102-3. See Annex A.

The “Data” element in the IIC-Response PDUs shall contain the SAP service primitives.

— MF-Command-confirm

— MN-Command-confirm

— MI-Command-confirm

— MF-Request-confirm

— MN-Request-confirm

— MI-Request-confirm

— MI-Get-confirm

— MI-Set-confirm