

DLMS Protocol Investigation

Bach Nguyen

Renesas Design Vietnam Co., Ltd.
Software Dept.

April 14, 2014

Rev. 3.00

Revision History

Rev.	Date	Description	
		Slide No.	Summary
1.00	July 10, 2013	All	Initialized revision

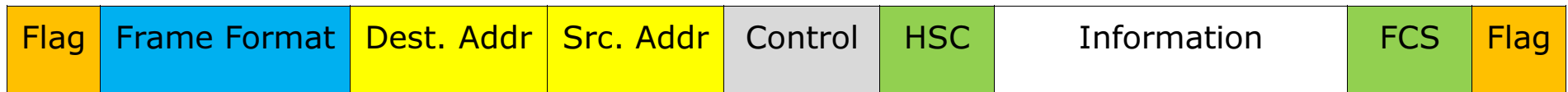
Outline

- HDLC Frame Type 3 ▶
- Command and response frame ▶
- APDU format ▶
 - AARQ APDU ▶
 - AARE APDU ▶
 - GET service ▶
 - SET service ▶
 - ACTION service ▶
 - Event notification service ▶
 - Appendix
 - Object List ▶
 - Selective access ▶
 - Cipher/Decipher ▶

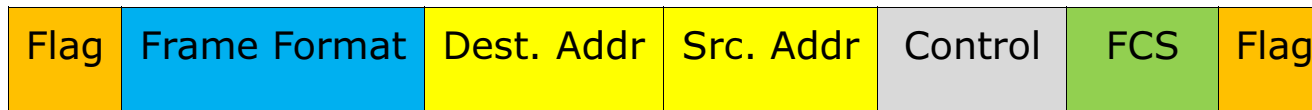
HDLC frame format type 3

HDLC frame format type 3

- HDLC Frame Type 3 as PDU Format as below:



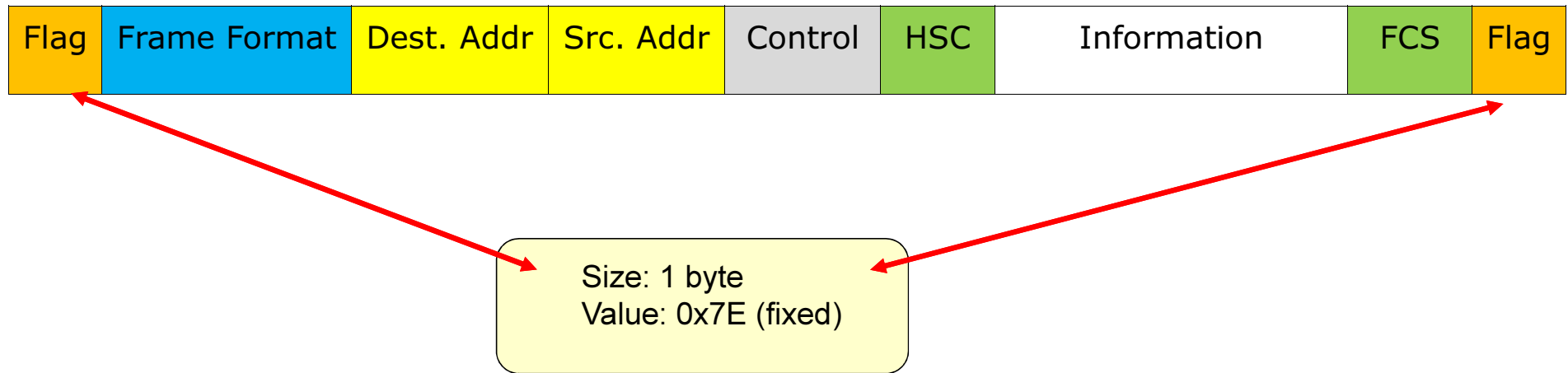
- If lack of Information filed (Ex: SNRM):



HDLC frame format type 3

Example:

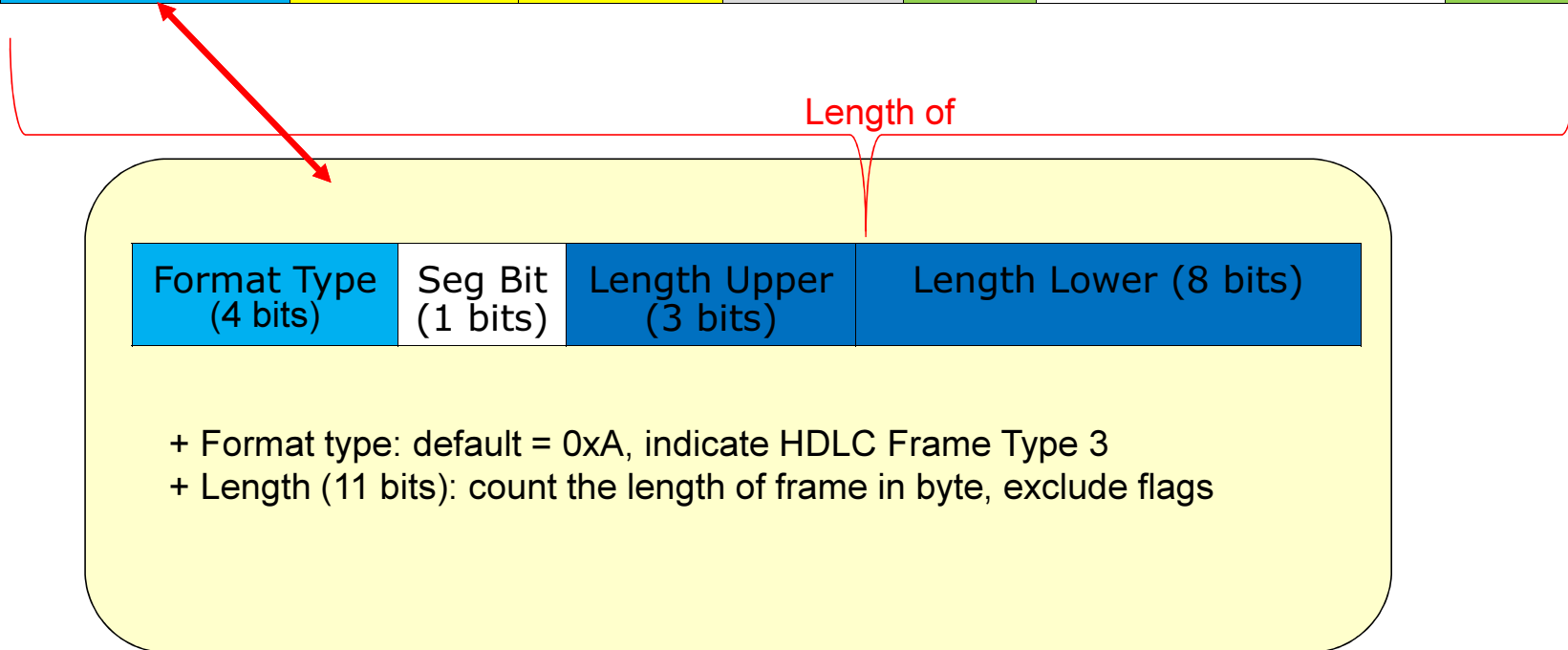
7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
FFFFD4C57E040E01000000065F1F040000301DFFFFD4C57E



HDLC frame format type 3

Example:

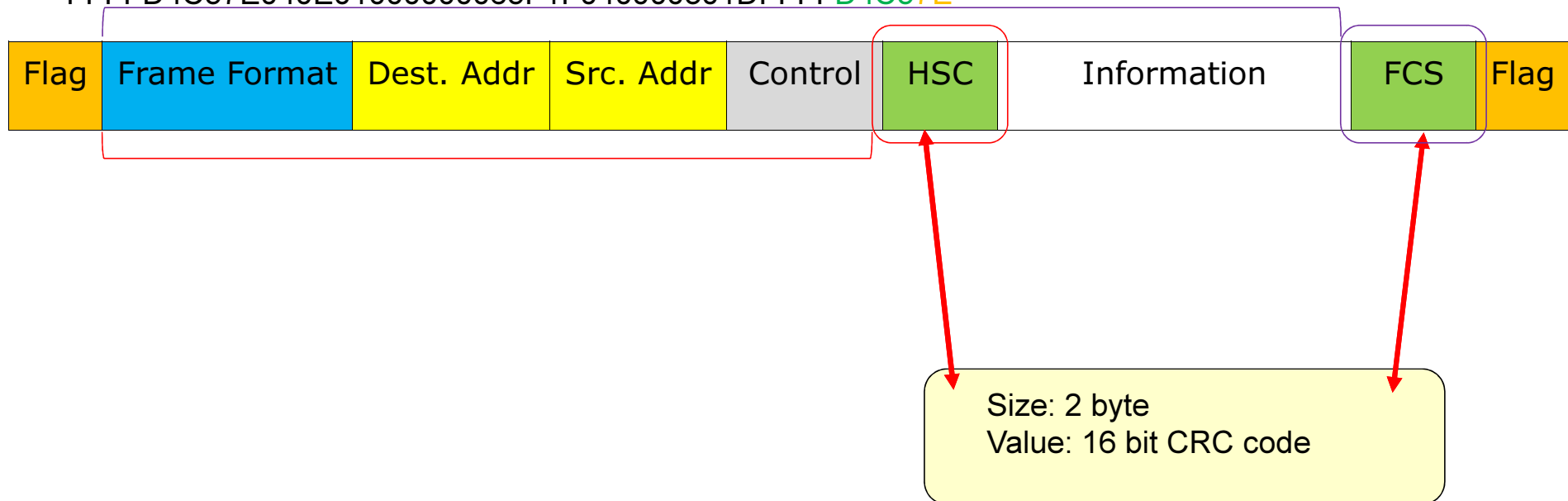
7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
FFFFD4C57E040E01000000065F1F040000301DFFFFD4C57E



HDLC frame format type 3

- Example:

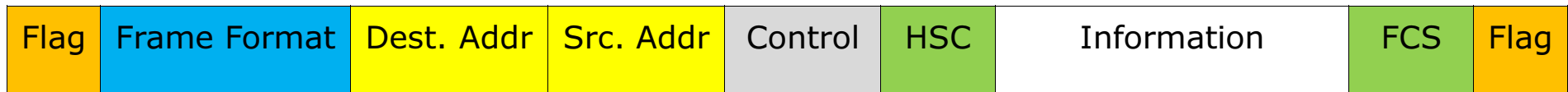
7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
FFFD4C57E040E01000000065F1F040000301DFFFFD4C57E



HDLC frame format type 3

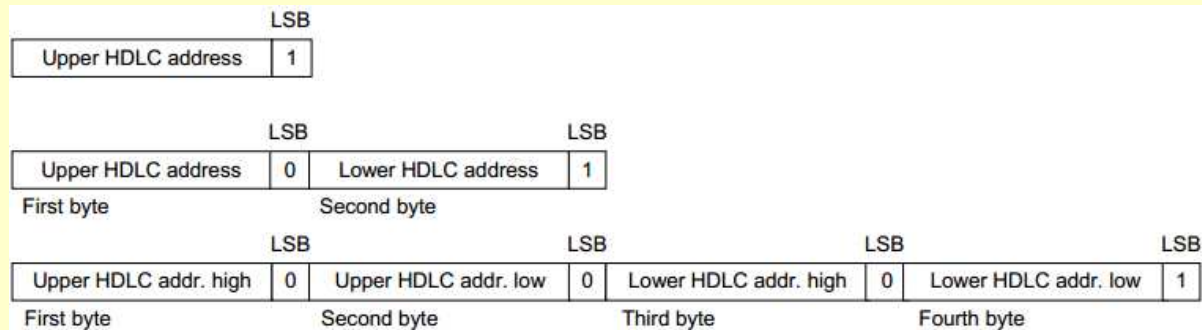
Example:

7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
 FFFFD4C57E040E01000000065F1F040000301DFFFFD4C57E



- + Size: server address: 1, 2 or 4 bytes
 - Sent by client: dest. addr,
 - Sent by server: src. addr)
- + Note
 - Upper HDLC address: Logical Device
 - Lower HDLC address: Physical Device
- + Ex: 00020023 ⇔ Upper: 0x0001 (Management Logical Device)
 Lower: 0x0011 (Physical Device address)

Detail format:



HDLC frame format type 3

- Example:

7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
FFFFD4C57E040E01000000065F1F040000301DFFFFD4C57E



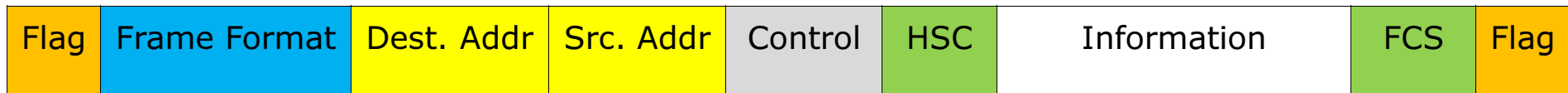
+ Size: Client address 1 byte
- Sent by client: src. addr,
- Sent by server: dest. addr

Ex: 0x21 ⇔ 0x10 (Public client)

HDLC frame format type 3

Example:

7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
 FFFFD4C57E040E01000000065F1F040000301DFFFFD4C57E



+ Size: 1 byte
 + Value:

N(R) (3 bits)	P/F (1bit)	N(S) (3 bits)	dummy (1 bit)
------------------	---------------	------------------	------------------

Note:

- + P for server, F for client (1/0 for enable/disable).
- + SNRM frame: 0x83 or 0x93.
- + DISC frame: 0x43 or 0x53.
- + UA frame: 0x63 or 0x73.
- + DM frame: 0x0F or 0x1F.
- + FRMR frame: 0x87 or 0x97.
- + UI frame: 0x03 or 0x13.
- + I frame: 0xXY (Y= 0, 2, 4, 6, 8, A, C, E).
- + RR frame: 0xX1.
- + RNR frame: 0xX5.

Command	Response	MSB				LSB			
		R	R	R	P/F	S	S	S	0
I	I	R	R	R	P/F	S	S	S	0
RR	RR	R	R	R	P/F	0	0	0	1
RNR	RNR	R	R	R	P/F	0	1	0	1
SNRM		1	0	0	P	0	0	1	1
DISC		0	1	0	P	0	0	1	1
	UA	0	1	1	F	0	0	1	1
	DM	0	0	0	F	1	1	1	1
	FRMR	1	0	0	F	0	1	1	1
UI	UI	0	0	0	P/F	0	0	1	1

HDLC frame format type 3

- Example:

7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
FFFFD4C57E040E01000000065F1F040000301DFFFFD4C57E



- + Size: max in [128: 2030] bytes.
- + With each frame type, it has different format.

Command and response frames

Command and response frames

HDLC frame type:

- ✚ SNRM: Set Normal Response Mode
- ✚ DISC: Disconnect
- ✚ DM: Disconnected mode response
- ✚ UI: Unnumbered Information
- ✚ UA: Unnumbered Acknowledge
- ✚ RR: Receive Ready
- ✚ RNR: Receive Not Ready
- ✚ FRMR: Frame Reject
- ✚ I: Information

Mode		NDM (Normal Disconnected Mode)				NRM (Normal Response Mode)		
response	command	SNRM	DISC	UI	UI(Empty)	I	RR	DISC
	accept	UA	DM	UA	UI	I	RR	UA
	reject						FRMR	
	Event Notification				UI	UI	UI	

Set normal response mode (SNRM) command

- Example:

7EA00802232193BD647E

Flag	Frame Format	Dest. Addr	Src. Addr	Control	FCS	Flag
------	--------------	------------	-----------	---------	-----	------

Unnumbered acknowledge (UA) response

Example:

7EA021210223738F72818014050200F2060200F20704000000010804000000010CF77E

Flag	Frame Format	Dest. Addr	Src. Addr	Control	HSC	Information	FCS	Flag
------	--------------	------------	-----------	---------	-----	-------------	-----	------

81	80	14	05	02	00	F2	06	02	00	F2	07	04	00	00	00	01	08	04	00	00	00	01
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

+ Size: any sequence of bytes.

+ Ex:

81H format id

80H group id

14H group length (20 octets)

05H parameter id (max info field length – transmit)

02H parameter length (2 octet)

00F2 parameter value

06H parameter id (max info field length – receive)

02H parameter length (2 octet)

00F2 parameter value

07H parameter id (window size, transmit)

04H parameter length (4 octets)

00H parameter value (high byte of value)

00H parameter value

00H parameter value

01H parameter value (low byte of value)

08H parameter id (window size, receive)

04H parameter length (4 octets)

00H parameter value (high byte of value)

00H parameter value

00H parameter value

01H parameter value (low byte of value)

Unnumbered information (UI) response

Example:

7EA00802234113E0857E

7EA02A030002002103CE49E6E700C2010000010000616100FF020408024D157E4D157E



LLC PDU format:



Size: 8 bits
Value:
0xE6 (command) or 0xE7(response)

Disconnect (DISC) command

• Example:

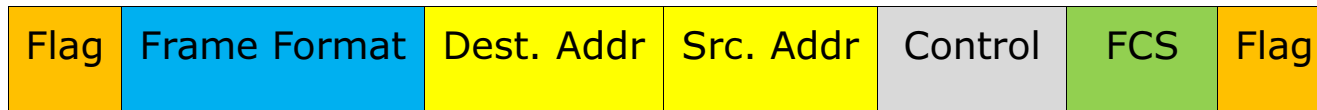
7EA00802232153B1A27E

Flag	Frame Format	Dest. Addr	Src. Addr	Control	FCS	Flag
------	--------------	------------	-----------	---------	-----	------

Disconnected mode (DM) response

• Example:

7EA0080223211FB1A27E



Receive Ready (RR) response

Example:

7EA008210223D162C57E

Flag	Frame Format	Dest. Addr	Src. Addr	Control	FCS	Flag
------	--------------	------------	-----------	---------	-----	------

RR shall be used by a data station to:

- + Indicate that it is ready to receive an I frame(s)
- + Acknowledge previously received I frames numbered up to $N(R) - 1$ inclusive.

When transmitted, the RR frame shall indicate the clearance of any busy condition that was initiated by the earlier transmission of an RNR frame by the same data station

Example:

7EA80E0223211E902101AC0A8038387E

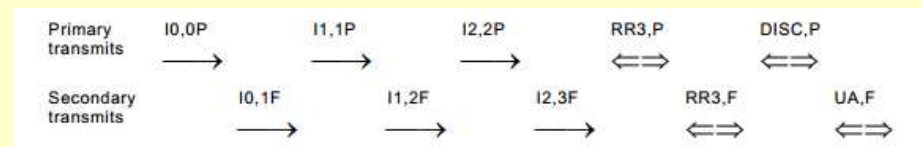
7EA008210223116E037E

7EA80E02232110EEC808414243B44B7E

7EA008210223316C227E

7EA80E02232112FCEB4445464712887E

7EA008210223516A417E



Frame reject (FRMR) response

• Example:

7EA00802232197B1A27E

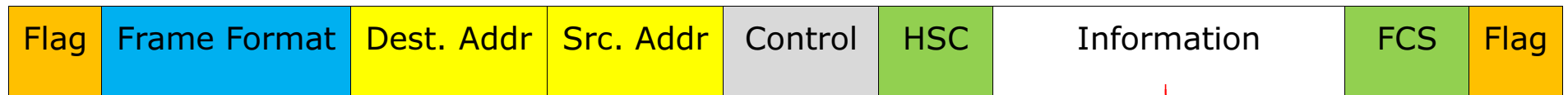
Flag	Frame Format	Dest. Addr	Src. Addr	Control	FCS	Flag
------	--------------	------------	-----------	---------	-----	------

Information (I) frame

- Information (I) frame

- Example:

7EA02E0002002321107ECB**E6**E600601DA109060760857405080101BE10040E01000000065F1F040000301D
FFFD4C57E040E01000000065F1F040000301DFFFD4C57E



LLC PDU format:



Size: 8 bits
Value: 0xE6

* 0xFF is used for broadcasting purposes

Information (I) frame

- Information (I) frame

- Example:

7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
FFFD4C57E040E01000000065F1F040000301DFFFD4C57E



LLC PDU format:



Size: 8 bits
Value:
0xE6 (command) or 0xE7(response)

Information (I) frame

- Information (I) frame

- Example:

7EA02E0002002321107ECBE6E600601DA109060760857405080101BE10040E01000000065F1F040000301D
FFFD4C57E040E01000000065F1F040000301DFFFD4C57E



LLC PDU format:



Size: 8 bits

Value:

- Its value is administered by the DLMS User Association. Currently, it must be set always to 0x00

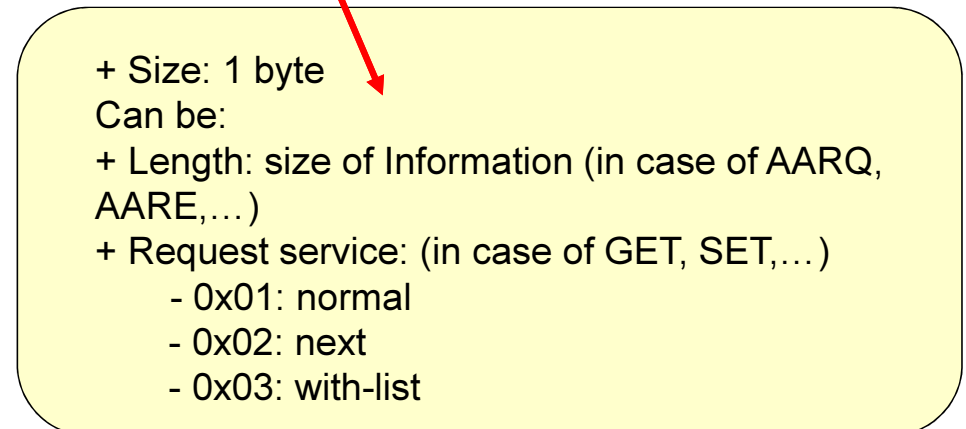
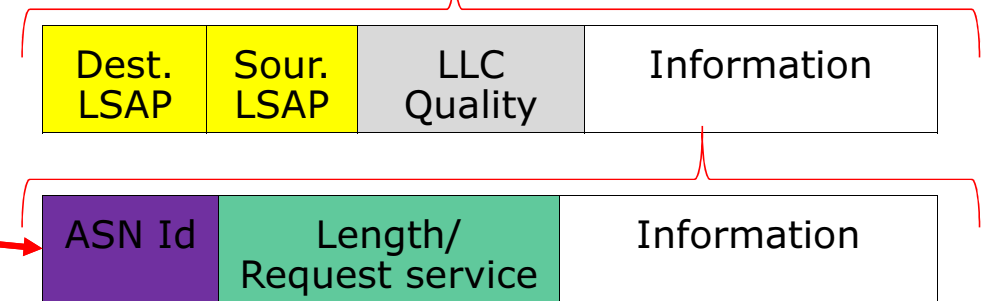
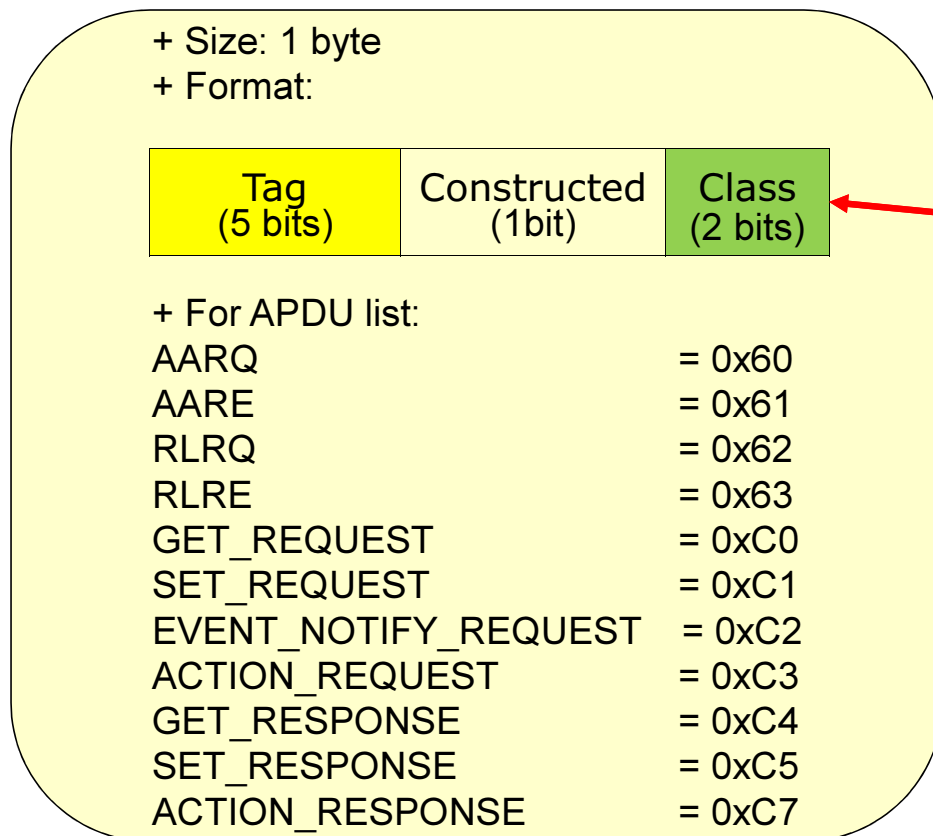
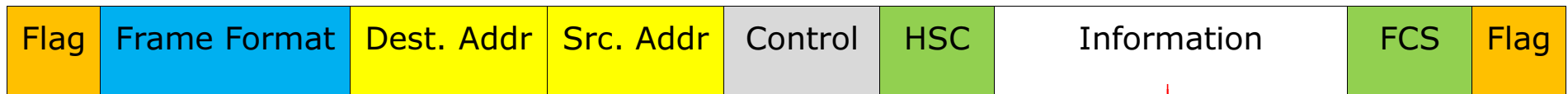
Information (I) frame

- Information (I) frame

- Example:

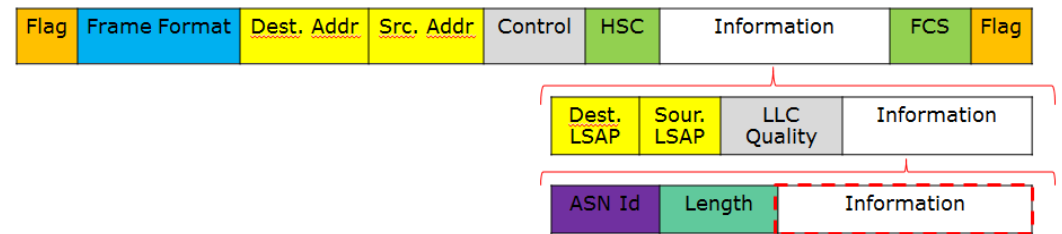
7EA02C02232110AF9FE6E600601DA109060760857405080101BE10040E01000000065F1F040000B81DFFFF
620D7E

7EA01A02232132F672E6E600C001C1000F0000280000FF020091537E



APDU format

AARQ APDU



AARQ APDU

- Information (I) frame

- Example:

7EA02C02232110AF9FE6E600601DA109060760857405080101BE10040E0100000065F1F040000B81DFFFF
620D7E

Application context name	ASCE requirement	Mechanism name	Calling authentication value	User information
--------------------------	------------------	----------------	------------------------------	------------------

+ BER encoding

+ This is fixed value:

A109060760857405080101

A1: The tag
09: Length
06: Choice for application-context-name
07: Length
60857405080101: Value

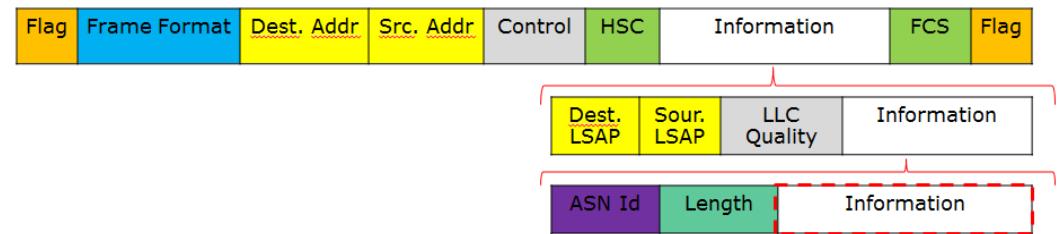
Table 45 – Use of ciphered / unciphered APDUs

Application context name	context_id	Unciphered APDUs	Ciphered APDUs
Logical_Name_Referencing_No_Ciphering ::=	context_id(1)	Yes	No
Short_Name_Referencing_No_Ciphering ::=	context_id(2)	Yes	No
Logical_Name_Referencing_With_Ciphering ::=	context_id(3)	Yes	Yes
Short_Name_Referencing_With_Ciphering ::=	context_id(4)	Yes	Yes

+ Note: ASN Id
AARQ

= 0x60

AARQ APDU



- Information (I) frame

- Example:

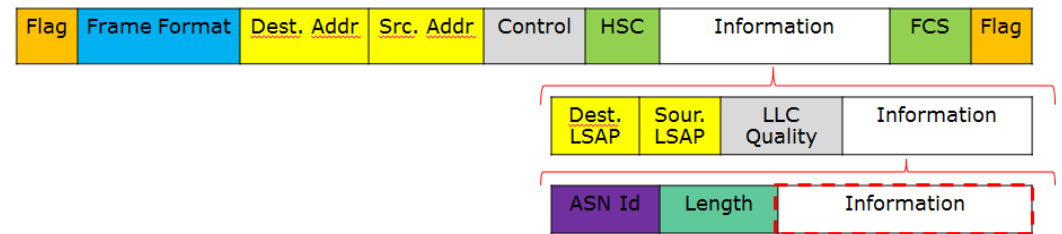
7EA02C02232110AF9FE6E600601DA109060760857405080101BE10040E01000000065F1F040000B81DFFFF620D7E

Application context name	ASCE requirement	Mechanism name	Calling authentication value	User information
--------------------------	------------------	----------------	------------------------------	------------------

- + BER encoding
- + For LLS and HLS, not used for “no security”
- + Ex:
8A020780
- 8A: The tag
- 02: Length
- 07: Number of unused bits in the last byte
- 80: Authentication functional unit (*)

Note: (*) only bit 0 set to 1 (indicating the requirement of the authentication functional unit)

AARQ APDU



- Information (I) frame

- Example:

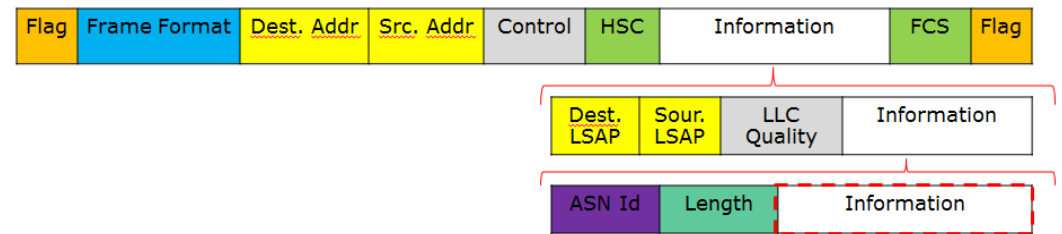
7EA02C02232110AF9FE6E600601DA109060760857405080101BE10040E01000000065F1F040000B81DFFFF620D7E

Application context name	ASCE requirement	Mechanism name	Calling authentication value	User information
--------------------------	------------------	----------------	------------------------------	------------------

+ BER encoding
 + For LLS and HLS, not used for "no security"
 + Ex:
 8B0760857405080201
 8B: The tag
 07: Length
 60857405080201: Value for LLS (*)

Note: (*) for HLS: 60857405080205
 - low-level-security-mechanism-name (1)
 - high-level-security-mechanism-name (5)

AARQ APDU



- Information (I) frame

- Example:

7EA02C02232110AF9FE6E600601DA109060760857405080101BE10040E01000000065F1F040000B81DFFFF620D7E

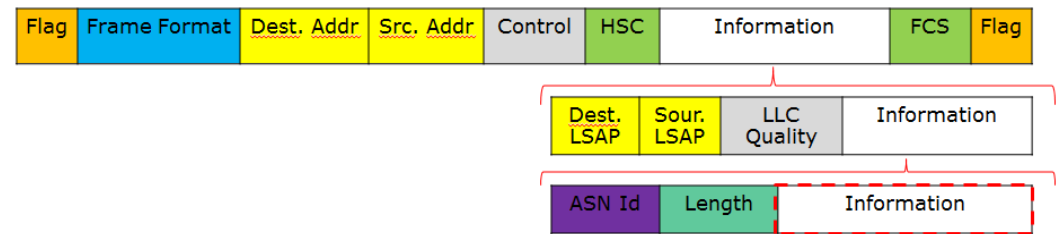
Application context name	ASCE requirement	Mechanism name	Calling authentication value	User information
--------------------------	------------------	----------------	------------------------------	------------------

+ BER encoding
 + For LLS and HLS, not used for “no security”
 + Ex:
 ACA080083132333435363738
 AC: The tag
 A0: Length
 80: Choice for Authentication-value
 08: Length of the Authentication value
 3132333435363738: Value for LLS (*)

Note: (*)

- In the case of LLS, the value of the Password “12345678” ⇔ 31 32 33 34 35 36 37 38
- In the case of HLS, the value of challenge CtoS “K56iVagY” ⇔ 4B 35 36 69 56 61 67 59

AARQ APDU



- Information (I) frame

- Example:

7EA02C02232110AF9FE6E600601DA109060760857405080101BE10040E01000000065F1F040000B81DFFFF620D7E

Application context name	ASCE requirement	Mechanism name	Calling authentication value	User information
--------------------------	------------------	----------------	------------------------------	------------------

+ BER encoding

+ Ex:

BE10040E01000000065F1F040000B81DFFFF

BE: The tag

10: Length

04: Choice for user info

0E: Length

01: Tag for xDLMS-Initiate request

00: Usage for dedicated-key component – not used

00: Usage for the response allowed component – not used

00: Usage of the proposed-quality-of-service component – not used

06: Proposed dlms version number 6

5F1F: Tag for conformance block

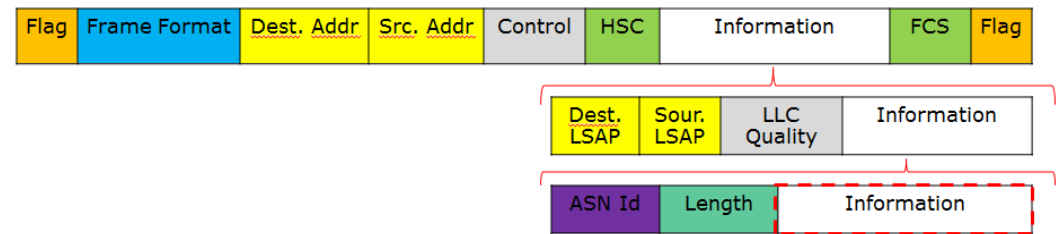
04: Length of the conformance block

00: Encoding the number of unused bits in the bit string

00B81D: conformance block

FFFF: Client-max-receive-pdu-size

AARQ APDU



- Information (I) frame

- Example:

7EA02C02232110AF9FE6E600601DA109060760857405080101BE10040E01000000065F1F040000B81DFFFF620D7E

Application context name	ASCE requirement	Mechanism name	Calling authentication value	User information
--------------------------	------------------	----------------	------------------------------	------------------

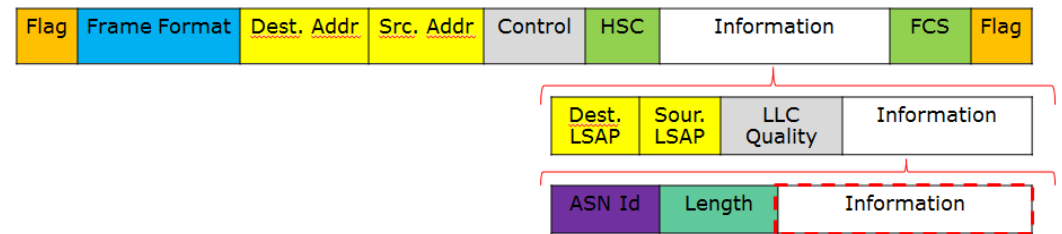
00B81D: conformance block

The bit is set when the corresponding service or functionality is available

reserved-zero (0),	block-transfer-with-set-or-write (12),
reserved-one (1),	block-transfer-with-action (13),
reserved-two (2),	multiple-references (14),
read (3),	information-report (15),
write (4),	reserved-sixteen (16),
unconfirmed-write (5),	reserved-seventeen (17),
reserved-six (6),	parameterized-access (18),
reserved-seven (7),	get (19),
attribute0-supported-with-set (8),	set (20),
priority-mgmt-supported (9),	selective-access (21),
attribute0-supported-with-get (10),	event-notification (22),
block-transfer-with-get-or-read (11),	action (23)

AARE APDU

AARE APDU



- Information (I) frame

- Example:

7EA05121022330E317E6E7006142A109060760857405080101A203020100A305A10302010088020780890760857405080201AA0A80084142434445464748BE10040E0800065F1F040000B81D00B4000704F97E

Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

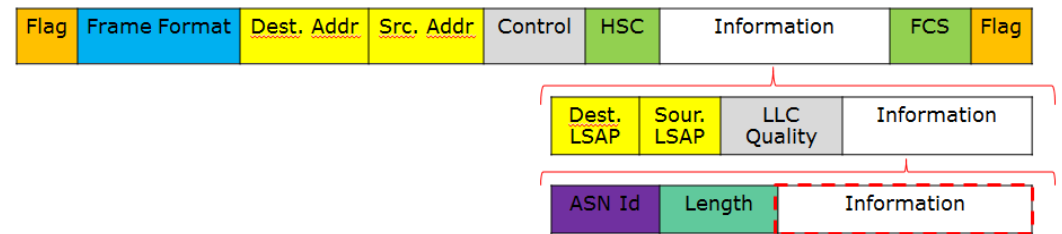
- + BER encoding
- + This is fixed value and same as AARQ

A109060760857405080101

A1: The tag
 09: Length
 06: Choice for application-context-name
 07: Length
 60857405080101: Value

+ Note: ASN Id
 AARE

= 0x61



AARE APDU

- Information (I) frame
- Example:

7EA05121022330E317E6E7006142A109060760857405080101A203020100A305A10302010088020780890760
857405080201AA0A80084142434445464748BE10040E0800065F1F040000B81D00B4000704F97E

Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

+ BER encoding

+ Ex:

A203020100

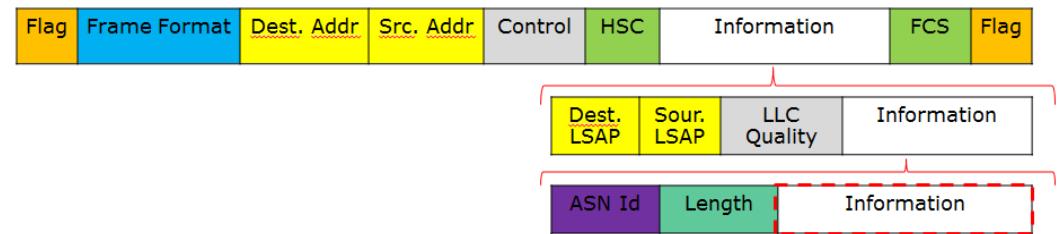
A2:	The tag
03:	Length
02:	Choice for result (Integer)
01:	Length of result value
00:	Accepted (*)

Note: (*)

- Success: 0

- Failure 1 and 2: 1

AARE APDU



- Information (I) frame

- Example:

7EA05121022330E317E6E7006142A109060760857405080101A203020100A305A10302010088020780890760857405080201AA0A80084142434445464748BE10040E0800065F1F040000B81D00B4000704F97E

Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

+ BER encoding

+ Ex:

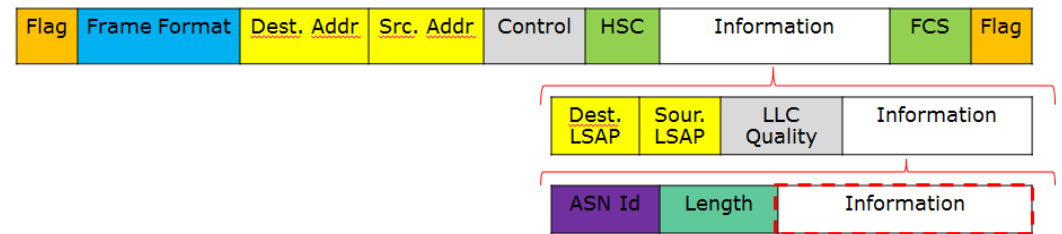
A305A103020100

A3: The tag
 05: Length
 A1: Tag for the acse-service-user CHOICE
 03: Length
 02: Choice for result (Integer)
 01: Length of result value
 00: Success (*)

Note: (*)

- Success: no security and LLS: 0
- Success: HLS security: 14 (0x0E)
- Failure 1: 2, application-context-name not supported
- Failure 2: 1, no-reason-given.

AARE APDU



- Information (I) frame

- Example:

7EA05121022330E317E6E7006142A109060760857405080101A203020100A305A10302010088020780890760
857405080201AA0A80084142434445464748BE10040E0800065F1F040000B81D00B4000704F97E

Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

+ BER encoding

+ For LLS and HLS, not used for “no security”

+ Ex:

88020780

88:

The tag

02:

Length

07:

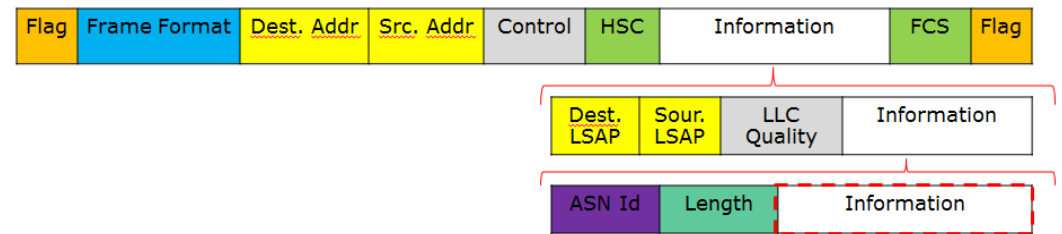
Number of unused bits in the last byte

80:

Authentication functional unit (*)

Note: (*) only bit 0 set to 1 (indicating the requirement of the authentication functional unit)

AARE APDU



- Information (I) frame

- Example:

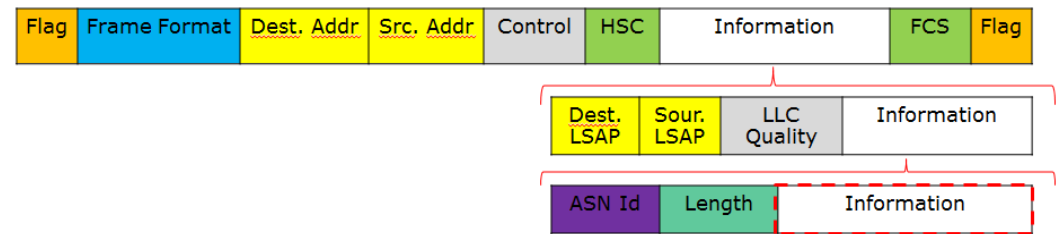
7EA05121022330E317E6E7006142A109060760857405080101A203020100A305A10302010088020780890760
857405080201AA0A80084142434445464748BE10040E0800065F1F040000B81D00B4000704F97E

Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

+ BER encoding
 + For LLS and HLS, not used for “no security”
 + Ex:
 890760857405080201
 89: The tag
 07: Length
 60857405080201: Value (*)

Note: (*) for HLS: 60857405080205
 - low-level-security-mechanism-name (1)
 - high-level-security-mechanism-name (5)

AARE APDU



- Information (I) frame

- Example:

7EA05121022330E317E6E7006142A109060760857405080101A203020100A305A10302010088020780890760
857405080201AA0A80084142434445464748BE10040E0800065F1F040000B81D00B4000704F97E

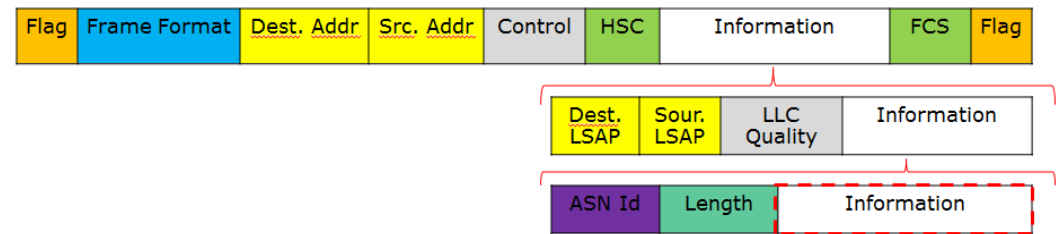
Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

+ BER encoding
 + For HLS only, not used for “no security” and LLS
 + Ex:
 AA0A80084142434445464748
 AA: The tag
 0A: Length
 80: Choice for Authentication-value
 08: Length of the Authentication value
 4142434445464748: Value (*)

Note: (*)

- LLS: encoding of the value of password
- HLS: encoding of the value of the challenge StoC

AARE APDU



- Information (I) frame
- Example:

7EA05121022330E317E6E7006142A109060760857405080101A203020100A305A10302010088020780890760857405080201AA0A80084142434445464748BE10040E0800065F1F040000B81D00B4000704F97E

Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

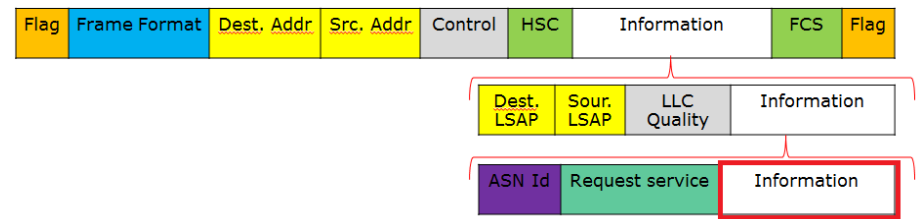
+ BER encoding

+ Ex:

BE10040E0800065F1F040000B81D00B40007

BE: The tag
 10: Length
 04: Choice for user info
 0E: Length
 08: Tag for xDLMS-Initiate response
 00: usage field of the negotiated-quality-of-service component
 06: negotiated dlms version number 6
 5F1F: Tag for conformance block
 04: Length of the conformance block
 00: Encoding the number of unused bits in the bit string
 00B81D: Negotiated conformance block
 00B4: Server-max-receive-pdu-size (180)
 0007: VAA name (0x0007 : LN referencing, 0xFA00: SN referencing.)
)

GET Service

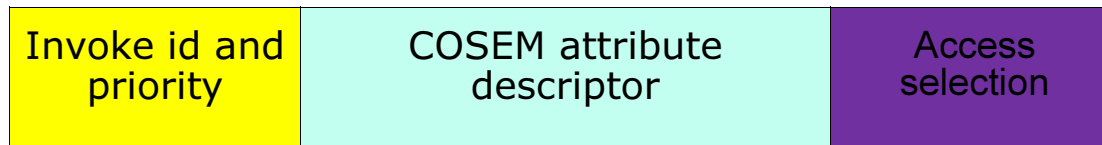


GET REQUEST APDU

Information (I) frame

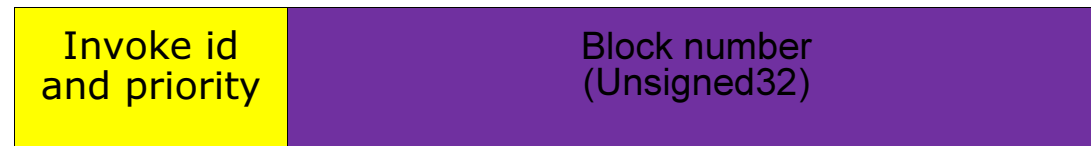
[Normal]

Ex: 7EA01A02232132F672E6E600C001C1000F0000280000FF020091537E



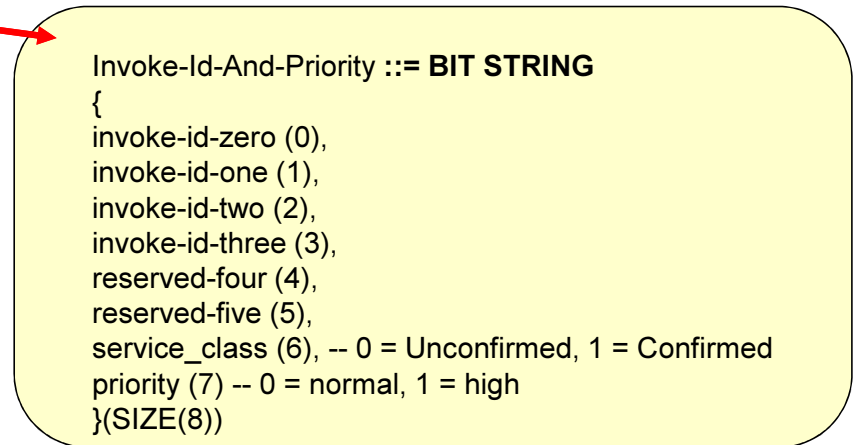
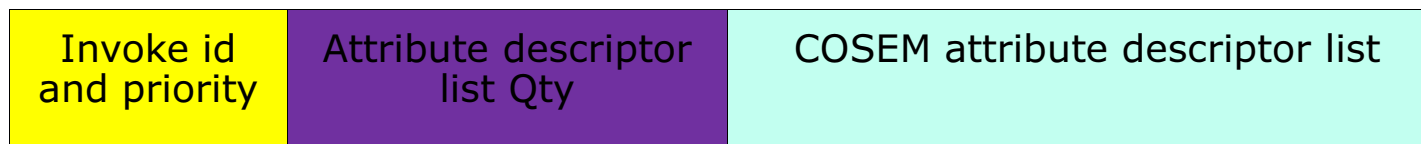
[Next]

Ex: 7EA014022321547E15E6E600C002C10000000151BE7E



[With list]

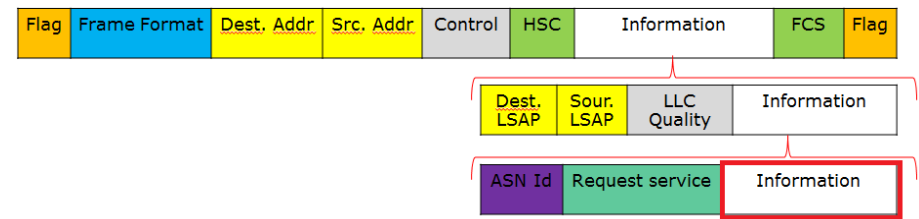
Ex: 7EA03902232132AB0FE6E600C003810400010000600100FF020000010000600101FF020000080000010000FF0400090C7E



+ Note: ASN Id
GET_RESQUEST

= 0xC0

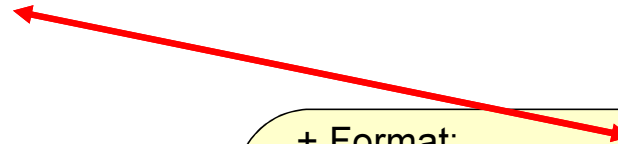
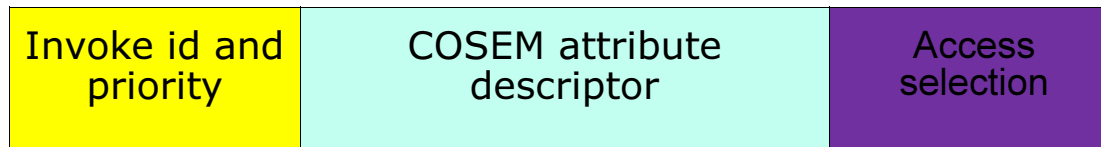
GET REQUEST APDU



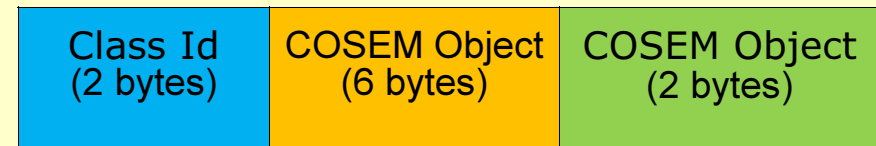
- Information (I) frame

- Example:

[Normal] 7EA01A02232132F672E6E600C001C1000F0000280000FF020091537E



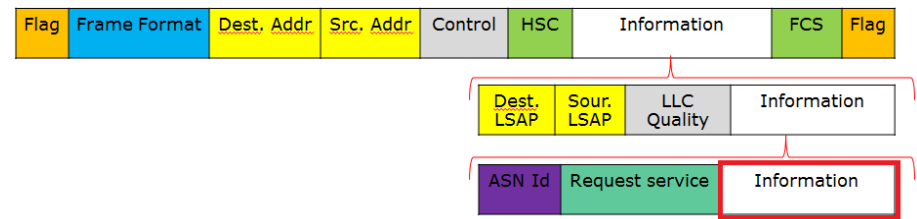
+ Format:



+ Ex:

000F0000280000FF02

⇔ Class 15, OBIS code {0,0,40,0,0,255}, attribute 2



GET RESPONSE APDU

- Information (I) frame
- Example:

[Normal]

7EA049410223527320E6E700C401C1000103020412000809060000010000FF0F021200000204120003090601001F5700FF0F02120000020412000309060100201B00FF0F0212000048757E

[Next]

7EA03E21022352B89CE6E700C402C100000000010027010502041200011100090600002A0000FF0202010202030F011601010002030F02160101000100807D7E



```

Get-Data-Result ::= CHOICE
{
  data [0] Data,
  data-access-result [1] Data-Access-Result
}
  
```

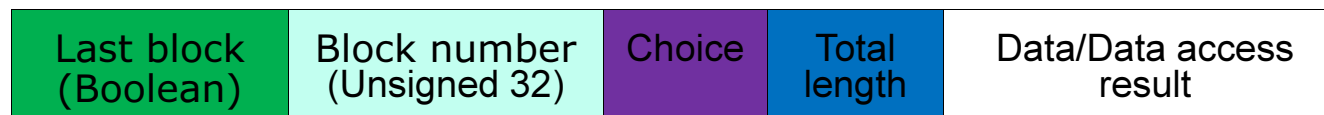
```

Data-Access-Result ::= ENUMERATED
{
  success (0),
  hardware-fault (1),
  temporary-failure (2),
  read-write-denied (3),
  object-undefined (4),
  object-class-inconsistent (9),
  object-unavailable (11),
  type-unmatched (12),
  scope-of-access-violated (13),
  data-block-unavailable (14),
  long-get-aborted (15),
  no-long-get-in-progress (16),
  long-set-aborted (17),
  no-long-set-in-progress (18),
  data-block-number-invalid (19),
  other-reason (250)
}
  
```

Get-Data-Result



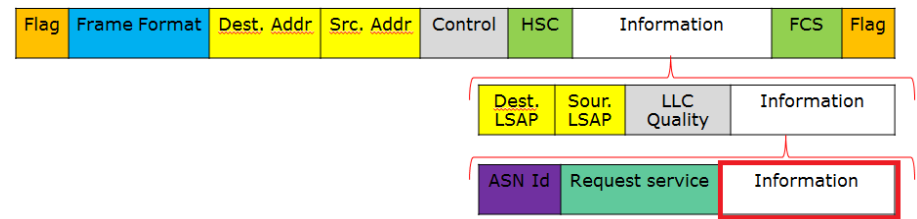
Data-Block-G



+ Note: ASN Id
GET_RESPONSE

= 0xC4

SET Service



SET REQUEST APDU

Information (I) frame

[Normal]

Ex: 7EA041022361321E06E6E600C1018100040100010600FF0000020509060100010600FF171234567802020F00161E09024142090C010203040506070809101112D5C67E

Invoke id and priority	COSEM attribute descriptor	Access selection	Value
------------------------	----------------------------	------------------	-------

[With-First-Datablock]

Ex:

7EA029022361328DFDE6E600C1028100040100010600FF000000000000000109020509060100010600DF447E

Invoke id and priority	COSEM attribute descriptor	Access selection	Last block (Boolean)	Data block	Value
------------------------	----------------------------	------------------	----------------------	------------	-------

[With-Datablock]

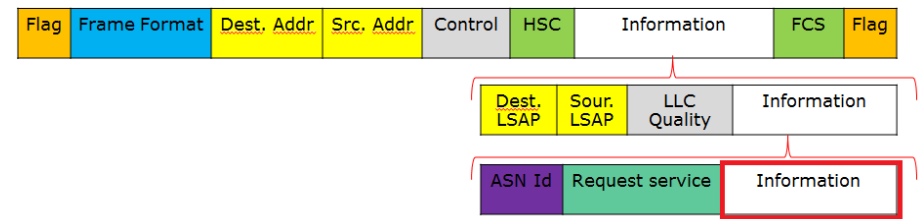
7EA01F02236154F414E6E600C10381000000000209FF171234567802020FAD0F7E

Invoke id and priority	Last block (Boolean)	Data block	Value
------------------------	----------------------	------------	-------

+ Note: ASN Id
SET_RESQUEST

= 0xC1

SET REQUEST APDU



Information (I) frame

[With-List]

Ex: 7EA06402236132DB40E6E600C104810400010000600100FF020000010000600101FF02000008000001000
0FF02000008000001000FF0500040905123456789009050987654321090C01020304050607080910111209
0C21222324252627282930313294027E

Invoke id and priority	Attribute descriptor list Qty	COSEM attribute descriptor list	ValueList Qty	Value list
------------------------------	----------------------------------	------------------------------------	------------------	------------

[-With-List-And-First-Datablock]

Ex: 7EA063022361320770E6E600C105810400010000600100FF020000010000600101FF02000008000001000
0FF02000008000001000FF050000000000010409280102030405060708091011121314151617181920212
223242526272829303132334E4C7E

Invoke id and priority	Attribute descriptor list Qty	COSEM attribute descriptor list	Last block (Boolean)	Data block	Value
------------------------------	----------------------------------	------------------------------------	-------------------------	------------	-------

+ Note: ASN Id
SET_RESQUEST

= 0xC1

SET RESPONSE APDU

- Information (I) frame

[Normal]

- Example:

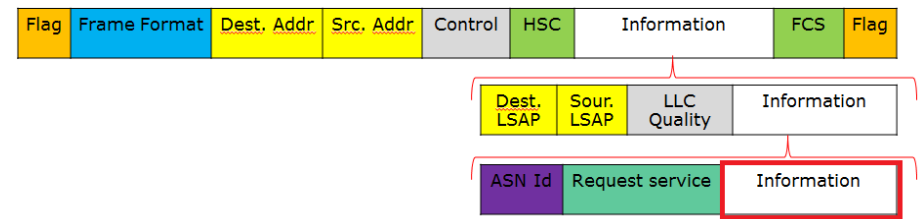
7EA0107595FC7105E6E700C501810036CF7E

Invoke id and priority	Result
------------------------	--------

[ForLastDataBlockWithList]

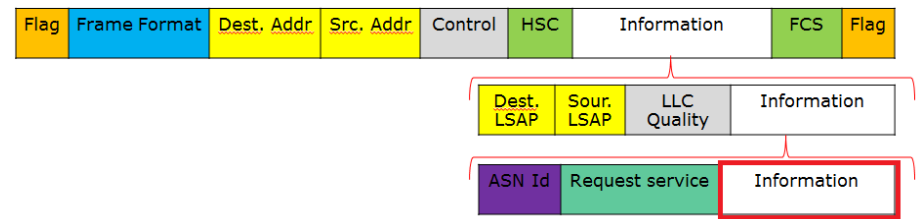
- Example:

7EA019610223B81692E6E700C50481040D0D00000000000432CD7E



+ Note: ASN Id
SET_RESPONSE = 0xC5

Action Service

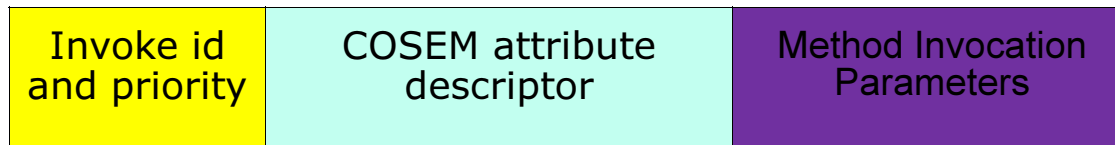


ACTION REQUEST APDU

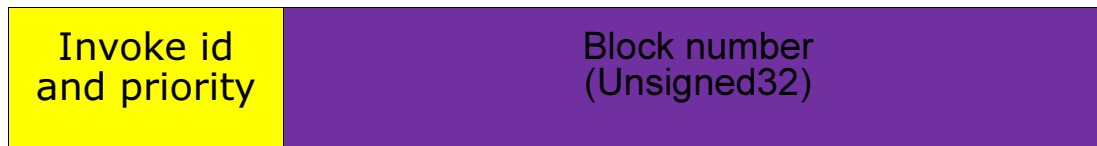
- Information (I) frame

[Normal]

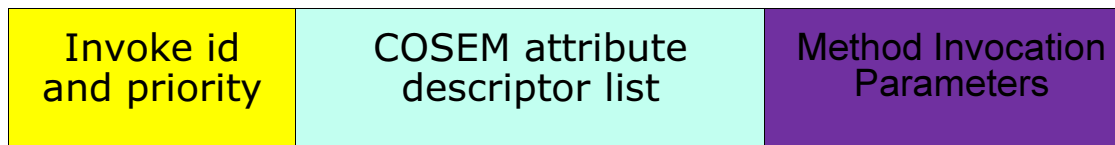
Ex: 7EA02C02236132D9DBE6E600C301C100F0000280003FF01010910884e6fdc35483f470C82D15D9C0343A34DF57E



[Next-Pblock]

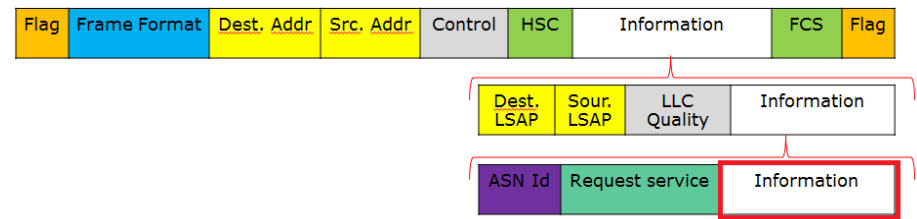


[With-List]



+ Note: ASN Id
ACTION_RESQUEST

= 0xC3



ACTION RESPONSE APDU

- Information (I) frame

[Normal]

- Example:

7EA02561022352A379E6E700C701C10001000910A0DD350ED17E8077D648EED3E201F48AB9AC7E



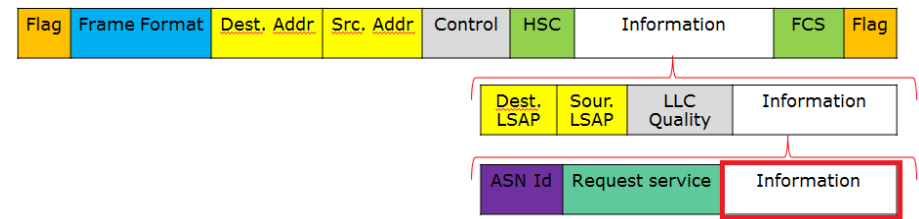
Single-response



+ Note: ASN Id
ACTION_RESPONSE = 0xC7

Event Notification Service

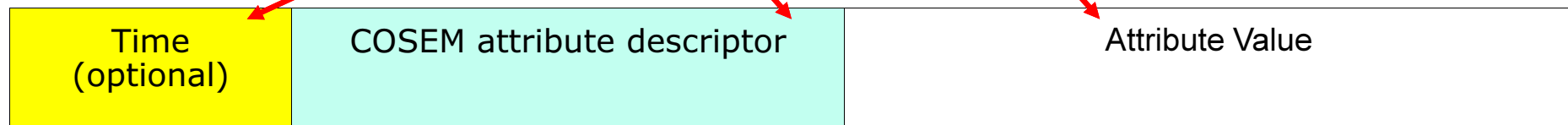
Event Notification REQUEST



Information (I) frame

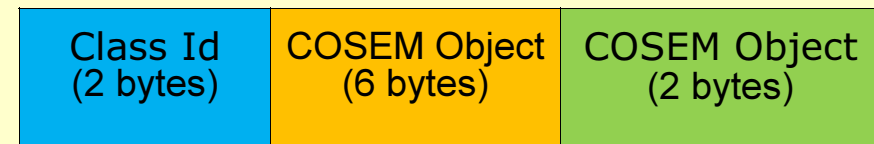
Ex:

7EA02A21022303E446E6E700C20101090C076C010107000235FF01A48000010000600B00FF0212000186A17E
7EA01C41022313C820E6E700C2010000010000600B02FF021200651B347E



Octet string

+ Format:



+ Ex:

000F0000280000FF02

⇔ Class 15, OBIS code {0,0,40,0,0,255}, attribute 2

EventNotificationRequest ::= **SEQUENCE**

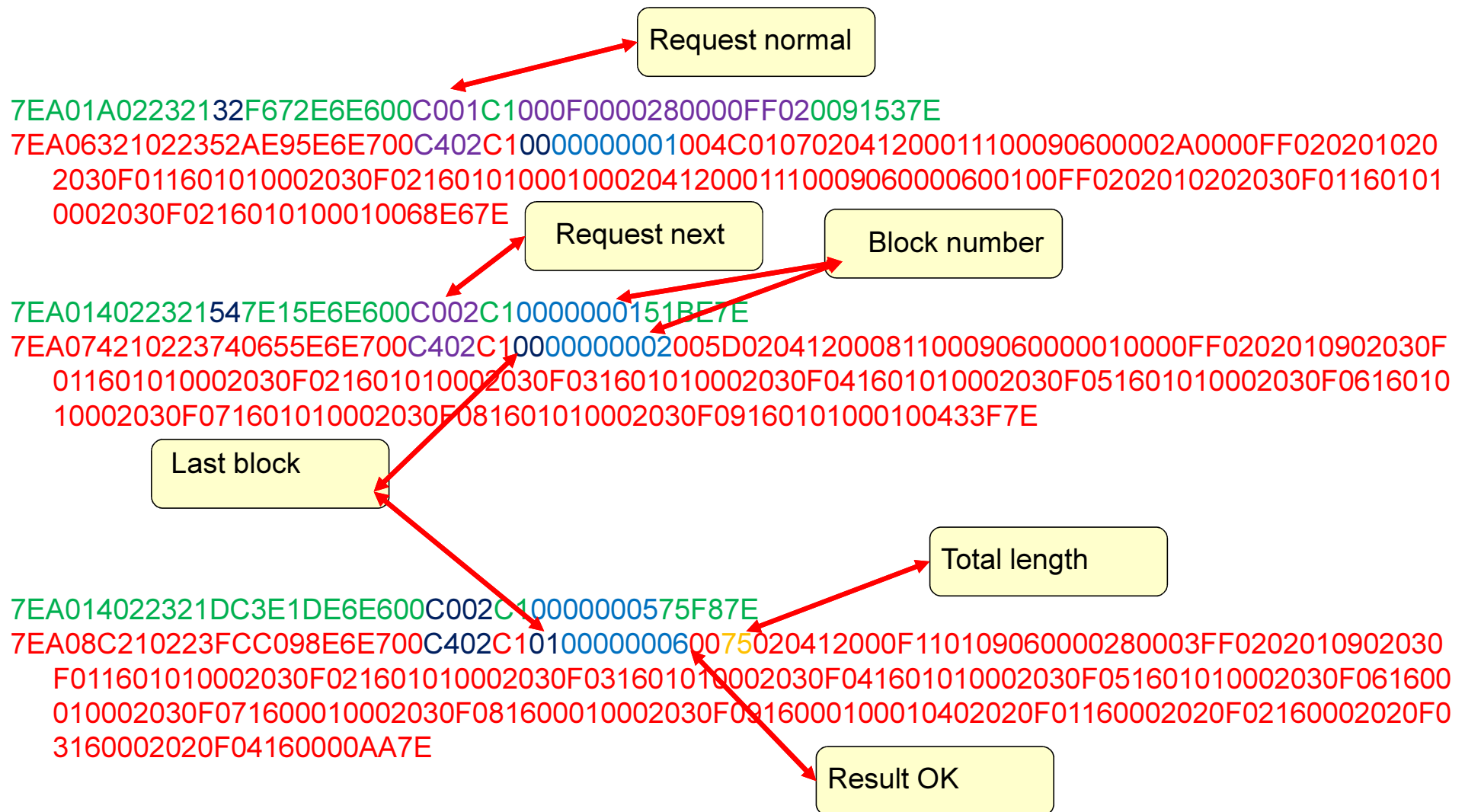
```
{
time OCTET STRING OPTIONAL,
cosem-attribute-descriptor Cosem-Attribute-Descriptor,
attribute-value Data
}
```

+ Note: ASN Id

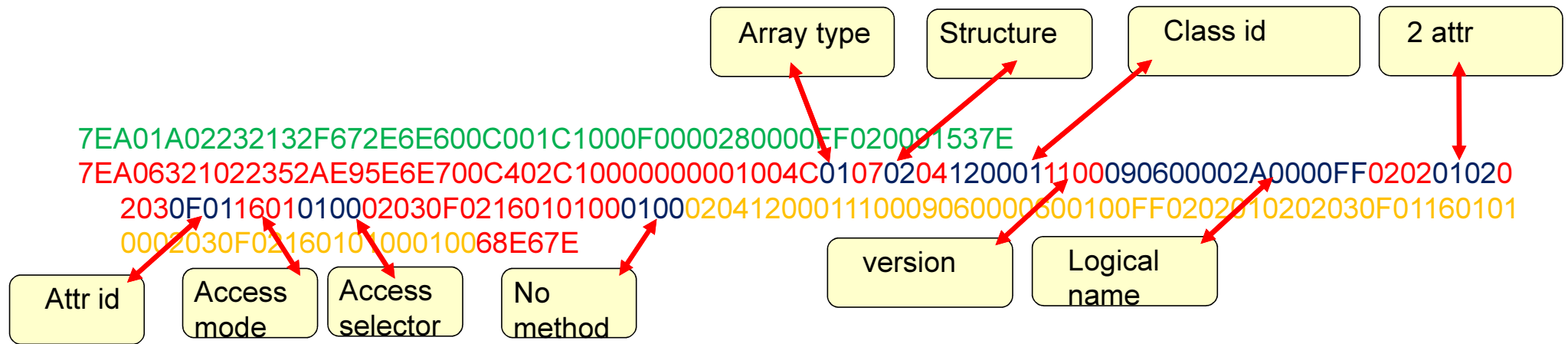
EVENT_NOTIFY_REQUEST = 0xC2

Object List

Object List



Object List

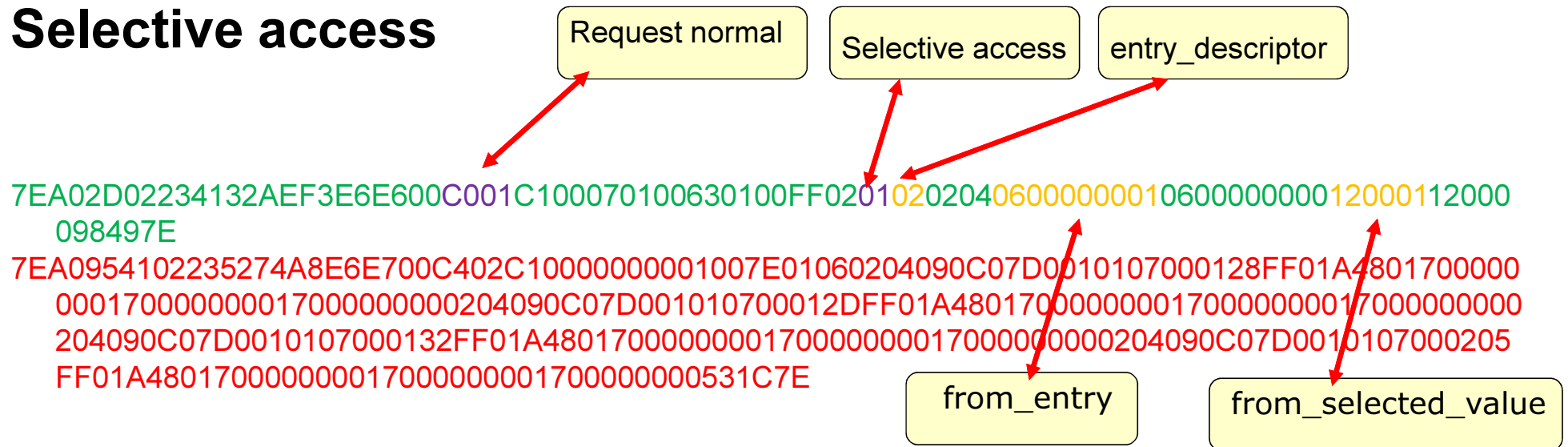


Object List

ATTR_TYPE_NULL_DATA	= 0
ATTR_TYPE_BOOLEAN	= 3
ATTR_TYPE_BIT_STRING	= 4
ATTR_TYPE_DOUBLE_LONG	= 5
ATTR_TYPE_DOUBLE_LONG_UNSIGNED	= 6
ATTR_TYPE_OCTET_STRING	= 9
ATTR_TYPE_VISIBLE_STRING	= 10
ATTR_TYPE_UTF8_STRING	= 12
ATTR_TYPE_BCD	= 13
ATTR_TYPE_INTEGER	= 15
ATTR_TYPE_LONG	= 16
ATTR_TYPE_UNSIGNED	= 17
ATTR_TYPE_LONG_UNSIGNED	= 18
ATTR_TYPE_LONG64	= 20
ATTR_TYPE_LONG64_UNSIGNED	= 21
ATTR_TYPE_ENUM	= 22
ATTR_TYPE_FLOAT32	= 23
ATTR_TYPE_FLOAT64	= 24
ATTR_TYPE_DATE_TIME	= 25
ATTR_TYPE_DATE	= 26
ATTR_TYPE_TIME	= 27
ATTR_TYPE_ARRAY	= 1
ATTR_TYPE_STRUCTURE	= 2
ATTR_TYPE_COMPACT_ARRAY	= 19

Selective access

Selective access



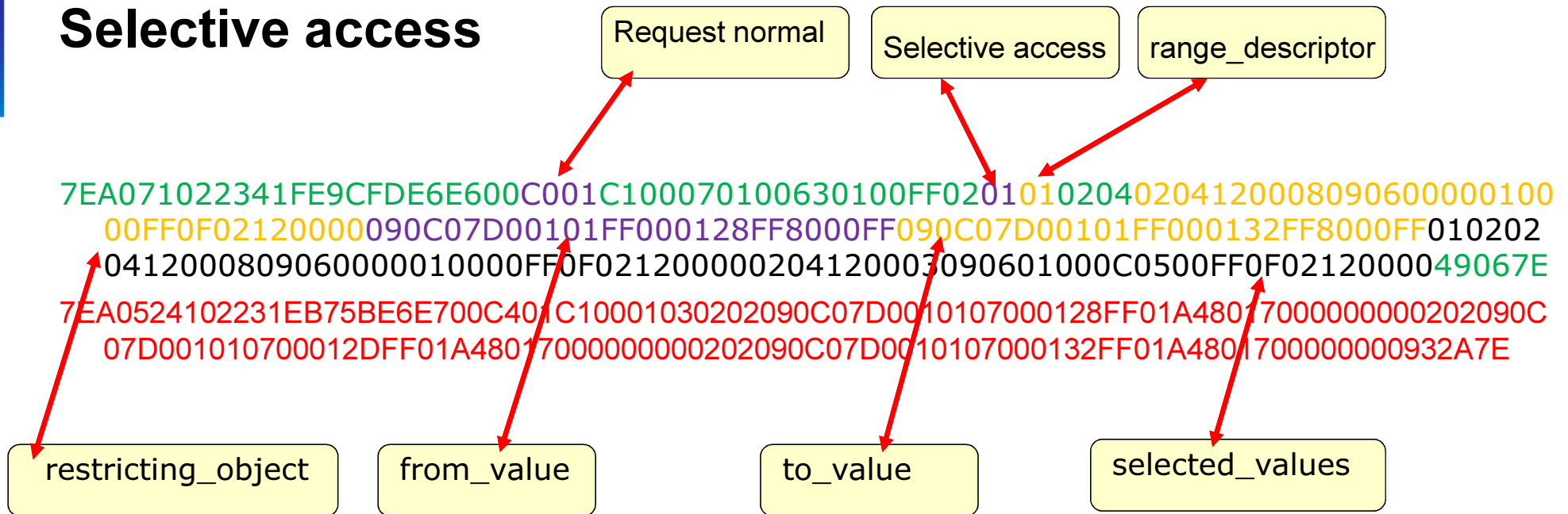
```
entry_descriptor ::= structure
{
  from_entry: double-long-unsigned first entry to retrieve,
  to_entry: double-long-unsigned last entry to retrieve
  from_selected_value: long-unsigned index of first value to retrieve,
  to_selected_value: long-unsigned index of last value to retrieve
}
```

Note:

to_entry == 0: highest possible entry,

to_selected_value == 0: highest possible selected_value

Selective access

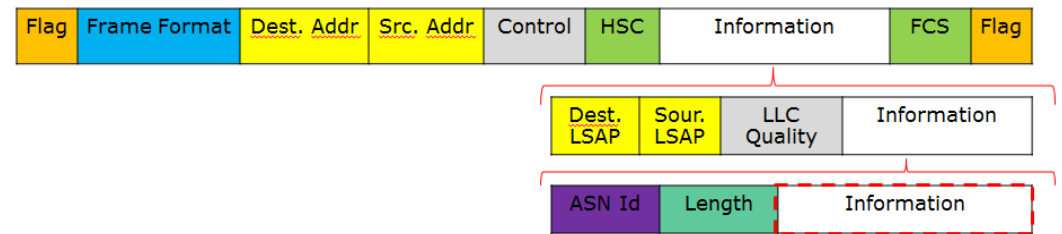


```
range_descriptor ::= structure
{
  restricting_object: capture_object_definition.
  from_value:   Oldest or smallest entry to retrieve
  to_value:    CHOICE
  selected_values: array capture_object_definition
}
```

AccessByRange(some entries 2 columns) From=01/01/2000 00:01:40, To=01/01/2000 00:01:50
 AccessByRange(some entries 2 columns) PASSED

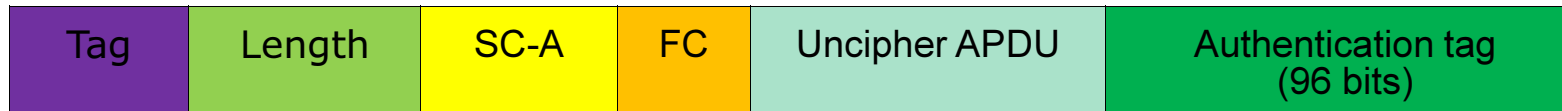
Cipher/Decipher

GET Cipher/decipher



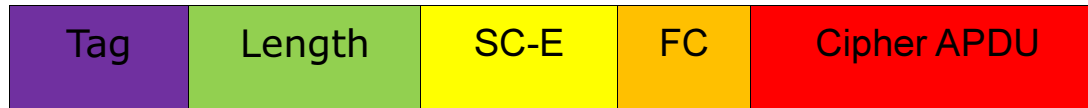
- [Ciphred APDU: Authentication only]

Ex: C81E1001234567C0010000080000010000FF020006725D910F9221D263877516



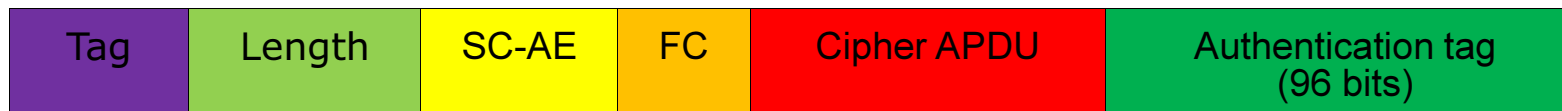
- [Ciphred APDU: Encryption only]

Ex: C8122001234567411312FF935A47566827C467BC

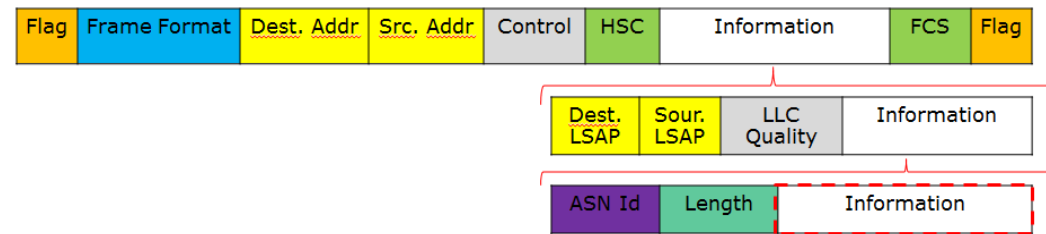


- [Ciphred APDU: Encryption + Authentication]

Ex: C81E3001234567411312FF935A47566827C467BC7D825C3BE4A77C3FCC056B6B



-xDLMS APDUs used with LN referencing
 -with global ciphering
 glo-get-request [200] IMPLICIT OCTET STRING,
 glo-set-request [201] IMPLICIT OCTET STRING,
 glo-event-notification-request [202] IMPLICIT OCTET STRING,
 glo-action-request [203] IMPLICIT OCTET STRING,
 glo-get-response [204] IMPLICIT OCTET STRING,
 glo-set-response [205] IMPLICIT OCTET STRING,
 glo-action-response [207] IMPLICIT OCTET STRING,



AARQ Cipher/decipher

- [Authenticated encryption of the xDLMs InitiateRequest APDU]

Ex: 7EA045022361101E29E6E600666A109060760857405080103A60A04084D4D4D0000BC614E8A0207808
 B0760857405080202AC0A80083132333435363738BE34043221303001234567801302FF8A7874133D414
 CED25B42534D28DB0047720606B175BD52211BE6841DB204D39EE6FDB8E35685555B07E

Application context name	calling-AP-title field	ASCE requirement	Mechanism name	Authentication value	User information
--------------------------	------------------------	------------------	----------------	----------------------	------------------

+ BER encoding

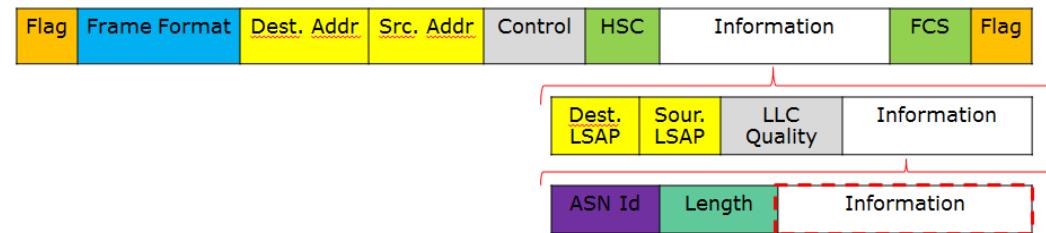
+ Ex:

A60A04084D4D4D0000BC614E

A6:	The tag
0A:	Length
04:	Choice for user info
08:	Length
4D4D4D0000BC614E	System title (64 bit)

with global ciphering

glo-initiateRequest [33] IMPLICIT OCTET STRING,
 glo-readRequest [37] IMPLICIT OCTET STRING,
 glo-writeRequest [38] IMPLICIT OCTET STRING,
 glo-initiateResponse [40] IMPLICIT OCTET STRING
 glo-readResponse [44] IMPLICIT OCTET STRING,



AARQ Cipher/decipher

- [Authenticated encryption of the xDLMS InitiateRequest APDU]

Ex: 7EA045022361101E29E6E60066A109060760857405080103A60A04084D4D4D0000BC614E8A0207808
 B0760857405080202AC0A80083132333435363738BE34043221303001234567801302FF8A7874133D414
 CED25B42534D28DB0047720606B175BD52211BE6841DB204D39EE6FDB8E35685555B07E

Application context name	calling-AP-title field	ASCE requirement	Mechanism name	Authentication value	User information
--------------------------	------------------------	------------------	----------------	----------------------	------------------

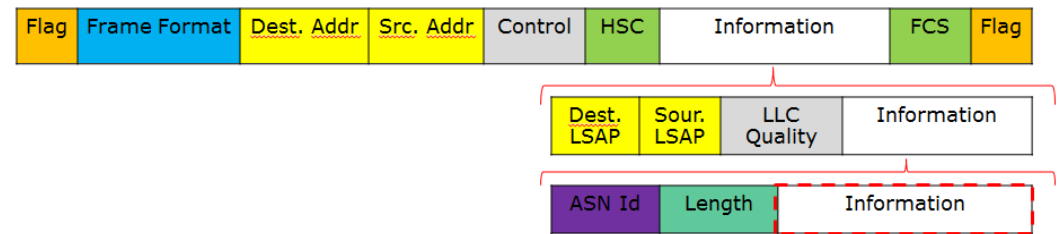
BE34043221303001234567801302FF8A7874133D414CED25B42534D28DB00
 47720606B175BD52211BE6841DB204D39EE6FDB8E356855

BE: The tag
 A0: Length
 04: Choice for user-information (OCTET STRING)
 32: Length of the Authentication value
 21303001234567801302FF8A7874133D414CED25B42534D28DB0047720606
 B175BD52211BE6841DB204D39EE6FDB8E356855
 : ciphered xDLMS InitiateRequest APDU (*)

Tag	Length	SC-AE	FC	Cipher APDU	Authentication tag (96 bits)
-----	--------	-------	----	-------------	------------------------------

with global ciphering
 glo-initiateRequest [33] IMPLICIT OCTET STRING,

AARE Cipher/decipher

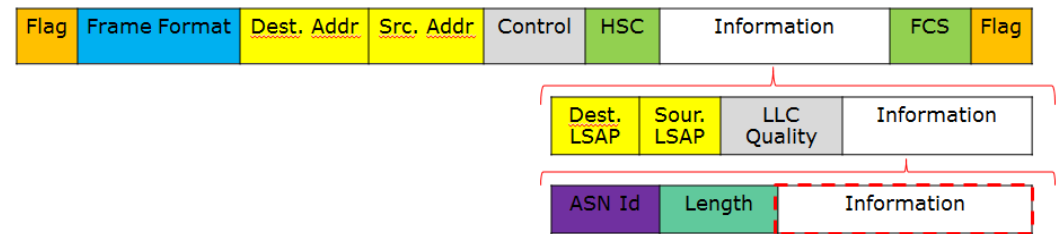


- [Authenticated encryption of xDLMS InitiateResponse APDU]

Ex: 7EA051610223305401E6E7006142A109060760857405080103A203020100A305A103020100A40A04084D4D4D0000BC614E BE230421281F3001234567891214A0845E475714383F65BC19745CA235906525E4F3E1C8936D207E

Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

+ BER encoding
 + For LLS and HLS, not used for “no security”
 + Ex:
 A40A04084D4D4D0000BC614
 A4: The tag
 0A: Length
 04: Choice for user-information (OCTET STRING)
 08: Length of the Authentication value
 4D4D4D0000BC614E System title (64 bit)



AARE Cipher/decipher

- [Authenticated encryption of xDLMS InitiateResponse APDU]

Ex: 7EA051610223305401E6E7006142A109060760857405080103A203020100A305A103020100A40A04084D
 4D4D0000BC614E BE230421281F3001234567891214A0845E475714383F65BC19745CA235906525E4F3
 E1C8936D207E

Application context name	Result	Result source diagnostic	Responder ASCE requirements	Mechanism name	Responding authentication value	User information
--------------------------	--------	--------------------------	-----------------------------	----------------	---------------------------------	------------------

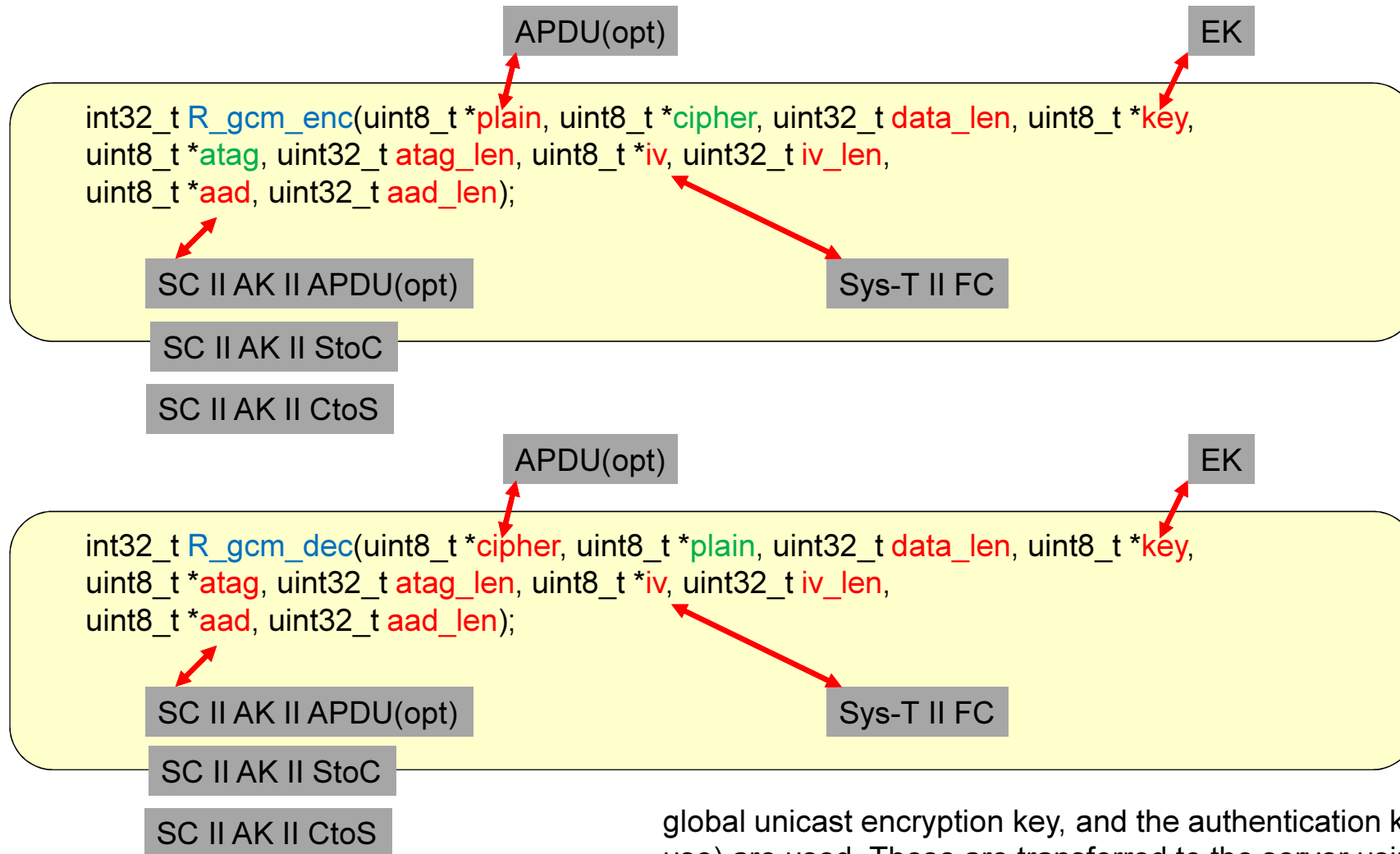
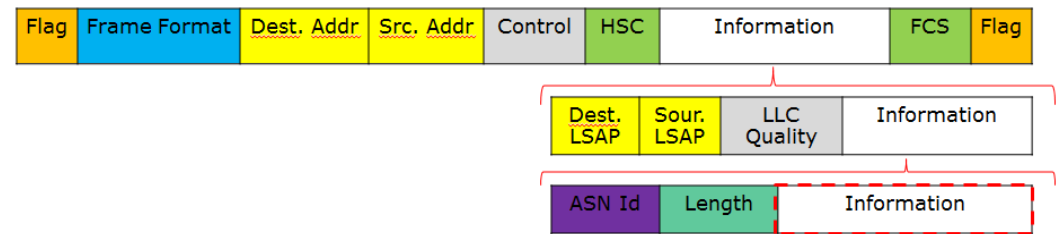
BE230421281F3001234567891214A0845E475714383F65BC19745CA235906525E4F3E1C893

BE: The tag
 23: Length
 04: Choice for user-information (OCTET STRING)
 21: Length of the Authentication value
 281F3001234567891214A0845E475714383F65BC19745CA235906525E4F3E1
 C893: ciphered xDLMS InitiateResponse APDU (*)

Tag	Length	SC-AE	FC	Cipher APDU	Authentication tag (96 bits)
-----	--------	-------	----	-------------	------------------------------

with global ciphering
 glo-initiateResponse [40] IMPLICIT OCTET STRING,

Cipher/decipher



global unicast encryption key, and the authentication key (if in use) are used. These are transferred to the server using the global_key_transfer method of the Security setup class



Renesas Design Vietnam Co., Ltd.