

# MOST

Media Oriented Systems Transport

Multimedia and Control  
Networking Technology

**MOST FBlock Template GeneralFBlock**

**Rev 3.0.7**

**01/2018**

**MOSTCO CONFIDENTIAL**

**See page 3 for the terms of disclosure**



## Legal Notice

### COPYRIGHT

© Copyright 1999 - 2018 MOST Cooperation. All rights reserved.

### LICENSE DISCLAIMER

Nothing on any MOST Cooperation Web Site, or in any MOST Cooperation document, shall be construed as conferring any license under any of the MOST Cooperation or its members or any third party's intellectual property rights, whether by estoppel, implication, or otherwise.

### CONTENT AND LIABILITY DISCLAIMER

MOST Cooperation or its members shall not be responsible for any errors or omissions contained at any MOST Cooperation Web Site, or in any MOST Cooperation document, and reserves the right to make changes without notice. Accordingly, all MOST Cooperation and third party information is provided "AS IS". In addition, MOST Cooperation or its members are not responsible for the content of any other Web Site linked to any MOST Cooperation Web Site. Links are provided as Internet navigation tools only.

MOST COOPERATION AND ITS MEMBERS DISCLAIM ALL WARRANTIES WITH REGARD TO THE INFORMATION (INCLUDING ANY SOFTWARE) PROVIDED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

In no event shall MOST Cooperation or its members be liable for any damages whatsoever, and in particular MOST Cooperation or its members shall not be liable for special, indirect, consequential, or incidental damages, or damages for lost profits, loss of revenue, or loss of use, arising out of or related to any MOST Cooperation Web Site, any MOST Cooperation document, or the information contained in it, whether such damages arise in contract, negligence, tort, under statute, in equity, at law or otherwise.

### FEEDBACK INFORMATION

Any information provided to MOST Cooperation in connection with any MOST Cooperation Web Site, or any MOST Cooperation document, shall be provided by the submitter and received by MOST Cooperation on a non-confidential basis. MOST Cooperation shall be free to use such information on an unrestricted basis.

### TRADEMARKS

MOST Cooperation and its members prohibit the unauthorized use of any of their trademarks. MOST Cooperation specifically prohibits the use of the MOST Cooperation LOGO unless the use is approved by the Steering Committee of MOST Cooperation.

### SUPPORT AND FURTHER INFORMATION

For more information on the MOST technology, please contact:

**MOST Cooperation**

Administration  
Emmy-Noether-Str. 14  
76131 Karlsruhe  
Germany

Tel: (+49) (0) 721 966 50 00

E-mail: [contact@mostcooperation.com](mailto:contact@mostcooperation.com)

Web: [www.mostcooperation.com](http://www.mostcooperation.com)



This Specification is Confidential Information of the MOST Cooperation. It may only be disclosed to member companies. Member companies wishing to discuss these Specifications with suppliers or other third parties must ensure that a commercially standard form of non-disclosure agreement has been previously executed by the party receiving such Specifications. Use of these Specifications may only be for purposes for which they are intended by the MOST Cooperation. Unauthorized use or disclosure is a violation of law.

© Copyright 1999 - 2018 MOST Cooperation  
All rights reserved

MOST is a registered trademark

<b>BIBLIOGRAPHY .....</b>	<b>6</b>
<b>DOCUMENT HISTORY .....</b>	<b>7</b>
<b>1 INTRODUCTION.....</b>	<b>12</b>
<b>2 FBLOCK TEMPLATE DEFINITION .....</b>	<b>12</b>
2.1 GeneralFBlock .....	12
2.1.1 FktIDs (0x000) .....	15
2.1.2 Notification (0x001) .....	16
2.1.3 NotificationCheck (0x002) .....	18
2.1.4 Version (0x010) .....	20
2.1.5 FBlockInfo (0x011) .....	21
2.1.6 DynArrayIns (0x080) .....	24
2.1.7 DynArrayDel (0x081) .....	26
2.1.8 MapIns (0x082) .....	27
2.1.9 MapDel (0x083) .....	29
2.1.10 CreateArrayWindow (0x090) .....	31
2.1.11 DestroyArrayWindow (0x091) .....	33
2.1.12 MoveArrayWindow (0x092) .....	34
2.1.13 SearchArrayWindow (0x093) .....	36
2.1.14 LongArrayInfo (0x094) .....	37
2.1.15 ArrayWindowIns (0x09A) .....	39
2.1.16 ArrayWindowDel (0x09B) .....	41
2.1.17 PowerDownDelay (0x0A0) .....	42
2.1.18 HDCP_ReceiverConnectedIndication (0x0C0) .....	43
2.1.19 HDCP_ReceiverDisconnectedIndication (0x0C1) .....	45
2.1.20 HDCP_Control (0x0C2) .....	47
2.1.21 HDCP_DecipherStatus (0x0C4) .....	49
2.1.22 HDCP_Assign (0x0C5) .....	50
2.1.23 SourceInfo (0x100) .....	51
2.1.24 Allocate (0x101) .....	57
2.1.25 DeAllocate (0x102) .....	59
2.1.26 SourceActivity (0x103) .....	60
2.1.27 SourceName (0x104) .....	61
2.1.28 AllocateExt (0x108) .....	62
2.1.29 SinkInfo (0x110) .....	65
2.1.30 Connect (0x111) .....	71
2.1.31 DisConnect (0x112) .....	73
2.1.32 Mute (0x113) .....	74
2.1.33 SinkName (0x114) .....	75
2.1.34 ConnectTo (0x115) .....	76
2.1.35 StreamDataInfo (0x116) .....	77
2.1.36 SinkRouting (0x117) .....	78
2.1.37 ConnectExt (0x118) .....	79
2.1.38 DTCP_StartProcess (0x120) .....	81
2.1.39 DTCP_Control (0x121) .....	82
2.1.40 DTCP_Status (0x122) .....	85
2.1.41 DTCP_CipherStatus (0x123) .....	88
2.1.42 DTCP_Info (0x124) .....	90
2.1.43 DTCP_ContentKeyProcess (0x125) .....	92
2.1.44 DTCP_InfoExt (0x126) .....	94
2.1.45 ScreenFormat (0x130) .....	96
2.1.46 VideoFrequency (0x131) .....	97
2.1.47 VideoNorm (0x132) .....	98
2.1.48 VideoSignalFormat (0x133) .....	99
2.1.49 VideoFormat (0x135) .....	100

2.1.50	DeckStatus (0x200) .....	101
2.1.51	TimePosition (0x201) .....	103
2.1.52	TrackPosition (0x202) .....	105
2.1.53	FramePosition (0x203) .....	106
2.1.54	TitlePosition (0x205) .....	107
2.1.55	ChapterPosition (0x206) .....	108
2.1.56	DeckStatusExt (0x207) .....	109
2.1.57	VideoInteraction (0x251) .....	112
2.1.58	PlayerRegion (0x270) .....	114
2.1.59	DeckEvent (0x430) .....	116
2.1.60	MediaEvent (0x431) .....	117
2.1.61	Random (0x450) .....	119
2.1.62	Scan (0x451) .....	120
2.1.63	Repeat (0x452) .....	121
2.1.64	NextTrackToPlay (0x453) .....	123
2.1.65	Deemphasis (0x454) .....	125
2.1.66	SlowFwSpeed (0x455) .....	126
2.1.67	SlowBwSpeed (0x456) .....	127
2.1.68	FastFwSpeed (0x457) .....	128
2.1.69	FastBwSpeed (0x458) .....	129

## Bibliography

All documents, which this MOST document has references to, are listed here with the actual revision this document is referring to.

Number	Document	Revision
[1]	MOST Specification	3.0

## Document History

### Changes GeneralFBlock FBlock Template 3.0.6 to GeneralFBlock FBlock Template 3.0.7

Change Ref.	FktID	Changes
3.0.7-001	0x0C3	Removed function HDCP_Status. (WGDA D158_6)

### Changes GeneralFBlock FBlock Template 3.0.5 to GeneralFBlock FBlock Template 3.0.6

Change Ref.	FktID	Changes
3.0.6-001	0x002	NotificationCheck: – Function specific error 0x08 "DeviceIDNotRegistered" and 0x20 "FBlockNotRegistered" deleted – Remodeled function specific Error.
3.0.6-002	0x100	SourceInfo: Marked DTCPiP as "deprecated".
3.0.6-003	0x101	Allocate: Remodeled function specific Error.
3.0.6-004	0x108	AllocateExt: Remodeled function specific Error.
3.0.6-005	0x110	SinkInfo: Marked DTCPiP as "deprecated".
3.0.6-006	0x121	DTCP_Control: Completely revised.
3.0.6-007	0x122	DTCP_Status: Completely revised.
3.0.6-008	0x123	DTCP_Cipher_Status: Changed parameter AvailableExchangeKeys from BitField to BoolField; added statement regarding SourceSinkNr.
3.0.6-009	0x124	DTCP_Info: Marked DTCPiP as "deprecated".
3.0.6-010	0x125	DTCP_ContentKeyProcess: Marked DTCPiP as "deprecated".
3.0.6-011	0x126	DTCP_InfoExt: Marked DTCPiP as "deprecated".
3.0.6-012	0x200	DeckStatus: Turned into an Unclassified Property to allow remodelling of function specific Error.
3.0.6-013	0x207	DeckStatusExt: Remodeled function specific Error.
3.0.6-014	0x251	VideoInteraction: Remodeled function specific Error.

### Changes GeneralFBlock FBlock Template 3.0.4 to GeneralFBlock FBlock Template 3.0.5

Change Ref.	FktID	Changes
3.0.5-001	–	Added GenericPCM_with_FrameRateMultiplier (0x12) to ContentType and TransmissionClass table.
3.0.5-002	0x011	FBlockInfo: Function has become conditional. It is not required for System FBlocks.
3.0.5-003	0x0C0	HDCP_ReceiverConnectedIndication: New function.
3.0.5-004	0x0C1	HDCP_ReceiverDisconnectedIndication: New function.
3.0.5-005	0x0C2	HDCP_Control: New function.
3.0.5-006	0x0C3	HDCP_Status: New function.
3.0.5-007	0x0C4	HDCP_DecipherStatus: New function.
3.0.5-008	0x0C5	HDCP_Assign: New function.
3.0.5-009	0x100	SourceInfo: Added GenericPCM_with_FrameRateMultiplier.
3.0.5-010	0x110	SinkInfo: Added GenericPCM_with_FrameRateMultiplier.
3.0.5-011	0x121	DTCP_Control: Modified description of parameter Subfunction.
3.0.5-012	0x126	DTCP_InfoExt: New function.

**Changes GeneralFBlock FBlock Template 3.0.3 to GeneralFBlock FBlock Template 3.0.4**

Change Ref.	FktID	Changes
3.0.4-001	0x100	SourceInfo: – Added MOST DTCP and HDCP to ContentProtection parameter. (WGDA 103_12_1) – Corrected symbolic name of Enum value for "Front Right of Center" in parameter AudioChannelName.
3.0.4-002	0x110	SinkInfo: – Added MOST DTCP and HDCP to ContentProtection parameter. (WGDA 103_12_1) – Corrected symbolic name of Enum value for "Front Right of Center" in parameter AudioChannelName.
3.0.4-003	0x120	DTCP_StartProcess: Removed OPTypes without SenderHandle. (WG Streaming)
3.0.4-004	0x121	DTCP_Control: Completely revised. (WGDA 103_17)
3.0.4-005	0x122	DTCP_Status: Completely revised. (WGDA 103_17)
3.0.4-006	0x125	DTCP_ContentKeyProcess: – Removed OPTypes without SenderHandle. (WG Streaming) – Corrected clerical error. MediaType parameter was missing for StartResultAck.
3.0.4-007	0x130	ScreenFormat: Adopted function from GeneralPlayer 2.5.1 and added symbolic names. (WGDA 109_6)
3.0.4-008	0x131	VideoFrequency: Adopted function from GeneralPlayer 2.5.1 and added symbolic names. (WGDA 109_6)
3.0.4-009	0x132	VideoNorm: Adopted function from GeneralPlayer 2.5.1 and added symbolic names. (WGDA 109_6)
3.0.4-010	0x133	VideoSignalFormat: Adopted function from GeneralPlayer 2.5.1 and added symbolic names. (WGDA 109_6)
3.0.4-011	0x135	VideoFormat: Adopted function from GeneralPlayer 2.5.1, extended with additional Enum values, and added symbolic names. (WGDA 109_6)
3.0.4-012	0x200	DeckStatus: Adopted function from GeneralPlayer 2.5.1, extended DeckStatus Enum, and added symbolic names. (WGDA 109_6)
3.0.4-013	0x201	TimePosition: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-014	0x202	TrackPosition: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-015	0x203	FramePosition: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-016	0x205	TitlePosition: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-017	0x206	ChapterPosition: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-018	0x207	DeckStatusExt (0x207): New function for extended DeckStatus. (WGDA 111)
3.0.4-019	0x251	– VideoInteraction: Adopted function from GeneralPlayer 2.5.1 and added OPTypes with SenderHandle. (WGDA 109_6) – Added Touch events.
3.0.4-020	0x270	PlayerRegion: Adopted function from GeneralPlayer 2.5.1 as Unclassified Property and changed parameter to Enum. (WGDA 109_6)
3.0.4-021	0x430	DeckEvent: Adopted function from GeneralPlayer 2.5.1, extended with additional Enum values, and added symbolic names. (WGDA 109_6)
3.0.4-022	0x431	MediaEvent: Adopted function from GeneralPlayer 2.5.1, extended with additional Enum values, and added symbolic names. (WGDA 109_6)
3.0.4-023	0x450	Random: Adopted function from GeneralPlayer 2.5.1 and added symbolic names. (WGDA 109_6)
3.0.4-024	0x451	Scan: Adopted function from GeneralPlayer 2.5.1 and added symbolic names. (WGDA 109_6)
3.0.4-025	0x452	Repeat: Adopted function from GeneralPlayer 2.5.1, extended with additional Enum values, and added symbolic names. (WGDA 109_6)
3.0.4-026	0x453	NextTrackToPlay: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-027	0x454	Deemphasis: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-028	0x455	SlowFwSpeed: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-029	0x456	SlowBwSpeed: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-030	0x457	FastFwSpeed: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)
3.0.4-030	0x458	FastBwSpeed: Adopted function from GeneralPlayer 2.5.1. (WGDA 109_6)



### Changes GeneralFBlock FBlock Template 3.0.2 to GeneralFBlock FBlock Template 3.0.3

Change Ref.	FktID	Changes
3.0.3-001	General	<ul style="list-style-type: none"> <li>– Corrected invalid exponent and step values.</li> <li>– Corrected invalid max. size and range values.</li> <li>– Combined duplicate parameters within single functions.</li> <li>– Corrected invalid parameter position values.</li> <li>– Fixed incorrect TEnumMax values.</li> <li>– Filled the SymbolicName attribute for all Enums.</li> </ul>
3.0.3-002	0x001	Notification: Removed blank after ClearAll Enum value of parameter Control.

### Changes GeneralFBlock FBlock Template 3.0.1 to GeneralFBlock FBlock Template 3.0.2

Change Ref.	FktID	Changes
3.0.2-001	General	– Added ContentType/TransmissionClass table to FBlock description.
3.0.2-002	0x100	<ul style="list-style-type: none"> <li>– SourceInfo: Distinguished between MOST50 and MOST150 ranges for BlockWidth, ClkSrcLabel, and ConnectionLabel.</li> <li>– Added ContentType SAD (0x08).</li> <li>– Added ContentType Ethernet (0x91).</li> <li>– Marked ContentType IP Stream (0x90) as deprecated.</li> </ul>
3.0.2-003	0x101	– Allocate: Distinguished between MOST50 and MOST150 ranges for BlockWidth and ConnectionLabel..
3.0.2-004	0x108	– AllocateExt: Distinguished between MOST50 and MOST150 ranges for BlockWidth, ClkSrcLabel, and ConnectionLabel.
3.0.2-005	0x110	<ul style="list-style-type: none"> <li>– SinkInfo: Distinguished between MOST50 and MOST150 ranges for BlockWidth and ConnectionLabel.</li> <li>– Added ContentType SAD (0x08).</li> <li>– Added ContentType Ethernet (0x91).</li> <li>– Marked ContentType IP Stream (0x90) as deprecated.</li> </ul>
3.0.2-006	0x111	– Connect: Distinguished between MOST50 and MOST150 ranges for BlockWidth and ConnectionLabel.
3.0.2-007	0x118	– Connect:Ext Distinguished between MOST50 and MOST150 ranges for BlockWidth, ClkSrcLabel, and ConnectionLabel.
3.0.2-008	0x124	– DTCP_Info: Added A/V Packetized DTCP-IP to PacketFormat Enum.
3.0.2-009	0x125	– DTCP_ContentKeyProcess: Added A/V Packetized DTCP-IP to PacketFormat Enum

### Changes GeneralFBlock FBlock 3.0.0 to GeneralFBlock FBlock 3.0.1

Change Ref.	FktID	Changes
3.0.1-001	General	<ul style="list-style-type: none"> <li>– Minor corrections</li> <li>– Removed "full range" from RangeMin attribute for unrestricted number parameters.</li> </ul>
3.0.1-002	0x090	CreateArrayWindow: – Added OPType ProcessingAck.
3.0.1-003	0x091	DestroyArrayWindow: – Added OPType ProcessingAck.
3.0.1-004	0x09A	ArrayWindowIns: <ul style="list-style-type: none"> <li>– Modified description of the Entry parameter.</li> <li>– Added parameter Taglist.</li> </ul>
3.0.1-005	0x100	SourceInfo: <ul style="list-style-type: none"> <li>– Added Phase Information (0x0F) to ContentType Enum.</li> <li>– Removed codes that are not used (0x80, 0x81, and 0x91) from ContentDescription parameter.</li> <li>– Added SpeedFactor to ContentDescription for ContentType 0x02.</li> <li>– Changed ContentType 0x90 to "IP Stream".</li> </ul>
3.0.1-006	0x101	Allocate: – Fixed clerical error; 0 is not a legal value for the ConnectionLabel.

Change Ref.	FktID	Changes
3.0.1-007	0x107	– Removed function SourceRouting.
3.0.1-008	0x108	– Added function AllocateExt.
3.0.1-009	0x110	SinkInfo: – Added Phase Information (0x0F) to ContentType Enum. – Removed codes that are not used (0x80, 0x81, and 0x91) from ContentDescription parameter. – Added SpeedFactor to ContentDescription for ContentType 0x02. – Changed ContentType 0x90 to “IP Stream”.
3.0.1-010	0x111	Connect: – Fixed clerical error; 0 is not a legal value for the ConnectionLabel.
3.0.1-011	0x116	StreamDataInfo: – Revised function and changed Occurrence attribute from optional to conditional.
3.0.1-012	0x118	– Added function ConnectExt.
3.0.1-013	0x124	DTCP_Info – Changed description of codes 0x02 and 0x03 of parameter PacketFormat.
3.0.1-014	0x125	DTCP_ContentKeyProcess – Changed description of codes 0x02 and 0x03 of parameter PacketFormat.

### Changes GeneralFBlock FBlock 2.5.1 - Speed Grade MOST50 to GeneralFBlock FBlock 3.0.0

Change Ref.	FktID	Changes
3.0.0-001	General	– Corrected ParamPos from 1 to 0 for parameterless OPTypes for reasons of consistency. – Unified descriptions of identical parameters and corrected clerical errors. – Added Occurrence attribute to all functions. – Replaced Channellist with BlockWidth, ConnectionLabel. – Removed SrcDelay parameter. – Removed SinkDelay parameter.
3.0.0-002	0x000	– Changed function class of FktIDs from Unclassified Method to Sequence Method.
3.0.0-003	0x001	– Removed remark that Notification messages must not be segmented. This is already described in more detail in the MOST Specification.
3.0.0-004	0x002	– Added function specific errors to NotificationCheck.
3.0.0-005	0x003	– Removed function AsyncControlSwitch.
3.0.0-006	0x011	– Added function FBlockInfo.
3.0.0-007	0x080	– Changed function class of DynArrayIns from Unclassified Method to Sequence Method. – Removed OPTYPE AbortAck from DynArrayIns.
3.0.0-008	0x081	– Changed function class of DynArrayDel from Unclassified Method to Sequence Method. – Removed OPTYPE AbortAck from DynArrayDel.
3.0.0-009	0x082	– Changed function class of MapIns from Unclassified Method to Sequence Method.
3.0.0-010	0x083	– Changed function class of MapDel from Unclassified Method to Sequence Method.
3.0.0-011	0x090	– Changed function class of CreateArrayWindow from Unclassified Method to Sequence Method. – Removed OPTYPE AbortAck from CreateArrayWindow.
3.0.0-012	0x091	– Changed function class of DestroyArrayWindow from Unclassified Method to Sequence Method. – Removed OPTYPE AbortAck from DestroyArrayWindow.
3.0.0-013	0x092	– Changed function class of MoveArrayWindow from Unclassified Method to Sequence Method. – Removed OPTYPE AbortAck from MoveArrayWindow.
3.0.0-014	0x093	– Changed function class of SearchArrayWindow from Unclassified Method to Sequence Method. – Removed OPTYPE AbortAck from SearchArrayWindow.

Change Ref.	FktID	Changes
3.0.0-015	0x09A	– Changed function class of ArrayWindowIns from Unclassified Method to Sequence Method.
3.0.0-016	0x09B	– Corrected description of parameter FktIDArrayWindow in ArrayWindowDel. – Changed SenderHandle type from Void to Unsigned Word in ArrayWindowDel.
3.0.0-017	0x100	– Completely revised SourceInfo.
3.0.0-018	0x101	– Changed function class of Allocate from Unclassified Method to Sequence Method. – Added SourceNr and BlockWidth as ErrorInfo in Allocate. – Added OPTypes StartResultAck, ErrorAck, ProcessingAck, and ResultAck to function Allocate.
3.0.0-019	0x102	– Changed function class of DeAllocate from Unclassified Method to Sequence Method. – Added OPTypes StartResultAck, ErrorAck, ProcessingAck, and ResultAck to function DeAllocate.
3.0.0-020	0x103	– Changed function class of SourceActivity from Unclassified Method to Sequence Method. – Added OPTypes StartResultAck, ErrorAck, ProcessingAck, and ResultAck to function SourceActivity.
3.0.0-021	0x104	– Removed limitation to 11 characters for SourceName parameter of the SourceName function.
3.0.0-022	0x110	– Completely revised SinkInfo.
3.0.0-023	0x111	– Changed function class of Connect from Unclassified Method to Sequence Method. – Added OPTypes StartResultAck, ErrorAck, ProcessingAck, and ResultAck to function Connect.
3.0.0-024	0x112	– Changed function class of Disconnect from Unclassified Method to Sequence Method. – Added OPTypes StartResultAck, ErrorAck, ProcessingAck, and ResultAck to function Disconnect.
3.0.0-025	0x114	– Removed limitation to 11 characters for SinkName parameter of the SinkName function.
3.0.0-026	0x115	– Changed function class of ConnectTo from Unclassified Method to Sequence Method. – Added OPTypes StartResultAck, ErrorAck, ProcessingAck, and ResultAck to function ConnectTo.
3.0.0-027	0x116	– Renamed SyncDataInfo to StreamDataInfo. – Replaced SinkCount and SourceCount with SinkNrList and SourceNrList in StreamDataInfo.
3.0.0-028	0x120	– Changed function class of DTCP_StartProcess from Unclassified Method to Sequence Method.
3.0.0-029	0x121	– Changed function class of DTCP_Control from Unclassified Method to Sequence Method.
3.0.0-030	0x122	– Changed function class of DTCP_Status from Unclassified Method to Sequence Method.
3.0.0-031	0x125	– Changed function class of DTCP_ContentKeyProcess from Unclassified Method to Sequence Method.

### Changes GeneralFBlock FBlock 2.5.1 - Speed Grade MOST25 to GeneralFBlock FBlock 2.5.1 – Speed Grade MOST50

Change Ref.	FktID	Changes
2.5.1-001	General	- Created new template for speed grade MOST50, based on GeneralFBlock 2.5.1 (speed grade MOST25).
2.5.1-002	General	- Added range information for parameters SrcDelay and SinkDelay (0-63).
2.5.1-003	General	- Modified parameter ChannelList so that it now consists of Blockwidth and ConnectionLabel.
2.5.1-004	General	- Modified parameter Blockwidth to either contain MOST50-specific information or 0x00.
2.5.1-005	General	- Added range information for parameters SourceNr and Resolution (1-255).
2.5.1-006	0x105	- Removed function SourceConnect (0x105), which is not used in MOST50.
2.5.1-007	0x106	- Removed function SourceDisconnect (0x106), which is not used in MOST50.

# 1 Introduction

This document contains the specification of an FBlock template. MOST FBlocks are standardized and maintained by MOST workgroup Device Architecture (WG\_DA). In order to speed up the process of making new Function Blocks available, every Function Block will be updated individually as required.

## 2 FBlock Template Definition

### 2.1 GeneralFBlock

This template describes functions that need to exist in every application (mandatory functions) and functions that only need to exist in applications with a data source and/or sink (optional functions). This section does not describe the functions of a specific FBlock, but contains a summary of functions that are optional and mandatory for FBlocks. The following FBlocks are no "real" applications and therefore are treated in a special way:

NetBlock, NetworkMaster, ConnectionMaster, PowerMaster and Diagnosis.

The following table illustrates the relation between ContentType and TransmissionClass. Asynchronous connections are not covered by the connection management; therefore, the ContentTypes of TransmissionClass "Asynchronous" are not provided by SourceInfo or SinkInfo.

ContentType	Description	Synchronous	DiscreteFrame Isochronous	Packetized Isochronous	QoS IP	Asynchronous
0x00	Audio	X	X			
0x02	SPDIF	X	X			
0x08	SAD	X				
0x0F	Phase Information		X			
0x10	GenericPCM	X	X			
0x12	GenericPCM_with_Frame RateMultiplier	X	X			
0x20	MPEG1_SystemStream	X		X		
0x21	MPEG1_ProgramStream	X		X		
0x22	MPEG1_TransportStream	X		X		
0x90	IP Stream				X	X
0x91	Ethernet				X	X
0xC0...0xEF	System Integrator specific	X dependent on the content	X dependent on the content	X dependent on the content	X dependent on the content	X dependent on the content
0xF0...0xFE	Supplier specific	X dependent on the content	X dependent on the content	X dependent on the content	X dependent on the content	X dependent on the content

Function Overview		
FktID	Name	Occurrence
0x000	<a href="#">FktIDs</a>	Mandatory
0x001	<a href="#">Notification</a>	Conditional
0x002	<a href="#">NotificationCheck</a>	Optional
0x010	<a href="#">Version</a>	Optional
0x011	<a href="#">FBlockInfo</a>	Conditional
0x080	<a href="#">DynArrayIns</a>	Optional
0x081	<a href="#">DynArrayDel</a>	Optional
0x082	<a href="#">MapIns</a>	Optional
0x083	<a href="#">MapDel</a>	Optional
0x090	<a href="#">CreateArrayWindow</a>	Conditional
0x091	<a href="#">DestroyArrayWindow</a>	Conditional
0x092	<a href="#">MoveArrayWindow</a>	Conditional
0x093	<a href="#">SearchArrayWindow</a>	Optional
0x094	<a href="#">LongArrayInfo</a>	Optional
0x09A	<a href="#">ArrayWindowIns</a>	Optional
0x09B	<a href="#">ArrayWindowDel</a>	Optional
0x0A0	<a href="#">PowerDownDelay</a>	Optional
0x0C0	<a href="#">HDCP_ReceiverConnectedIndication</a>	Conditional
0x0C1	<a href="#">HDCP_ReceiverDisconnectedIndication</a>	Conditional
0x0C2	<a href="#">HDCP_Control</a>	Conditional
0x0C4	<a href="#">HDCP_DecipherStatus</a>	Conditional
0x0C5	<a href="#">HDCP_Assign</a>	Conditional
0x100	<a href="#">SourceInfo</a>	Conditional
0x101	<a href="#">Allocate</a>	Conditional
0x102	<a href="#">DeAllocate</a>	Conditional
0x103	<a href="#">SourceActivity</a>	Optional
0x104	<a href="#">SourceName</a>	Optional
0x108	<a href="#">AllocateExt</a>	Optional
0x110	<a href="#">SinkInfo</a>	Conditional
0x111	<a href="#">Connect</a>	Conditional
0x112	<a href="#">DisConnect</a>	Conditional
0x113	<a href="#">Mute</a>	Optional
0x114	<a href="#">SinkName</a>	Optional
0x115	<a href="#">ConnectTo</a>	Optional
0x116	<a href="#">StreamDataInfo</a>	Conditional
0x117	<a href="#">SinkRouting</a>	Optional
0x118	<a href="#">ConnectExt</a>	Optional
0x120	<a href="#">DTCP_StartProcess</a>	Conditional
0x121	<a href="#">DTCP_Control</a>	Conditional
0x122	<a href="#">DTCP_Status</a>	Conditional
0x123	<a href="#">DTCP_CipherStatus</a>	Conditional
0x124	<a href="#">DTCP_Info</a>	Optional
0x125	<a href="#">DTCP_ContentKeyProcess</a>	Conditional
0x126	<a href="#">DTCP_InfoExt</a>	Conditional
0x130	<a href="#">ScreenFormat</a>	Optional
0x131	<a href="#">VideoFrequency</a>	Optional
0x132	<a href="#">VideoNorm</a>	Optional
0x133	<a href="#">VideoSignalFormat</a>	Optional
0x135	<a href="#">VideoFormat</a>	Optional
0x200	<a href="#">DeckStatus</a>	Optional
0x201	<a href="#">TimePosition</a>	Optional
0x202	<a href="#">TrackPosition</a>	Optional
0x203	<a href="#">FramePosition</a>	Optional
0x205	<a href="#">TitlePosition</a>	Optional

Function Overview		
0x206	<a href="#">ChapterPosition</a>	Optional
0x207	<a href="#">DeckStatusExt</a>	Optional
0x251	<a href="#">VideoInteraction</a>	Optional
0x270	<a href="#">PlayerRegion</a>	Optional
0x430	<a href="#">DeckEvent</a>	Optional
0x431	<a href="#">MediaEvent</a>	Optional
0x450	<a href="#">Random</a>	Optional
0x451	<a href="#">Scan</a>	Optional
0x452	<a href="#">Repeat</a>	Optional
0x453	<a href="#">NextTrackToPlay</a>	Optional
0x454	<a href="#">Deemphasis</a>	Optional
0x455	<a href="#">SlowFwSpeed</a>	Optional
0x456	<a href="#">SlowBwSpeed</a>	Optional
0x457	<a href="#">FastFwSpeed</a>	Optional
0x458	<a href="#">FastBwSpeed</a>	Optional

## 2.1.1 FktIDs (0x000)

Occurrence: Mandatory

This function is mandatory for every FBlock with one exception, the NetBlock. With the property FktIDs, the functions of an FBlock may be queried.

### 2.1.1.1 Format of Function

**Function classes:** Container

FBlock	Function	OPType	Parameter
GeneralFBlock	FktIDs (0x000)	Get	-
		Status	<a href="#">BitField</a>
		Error	ErrorCode, ErrorInfo

### 2.1.1.2 Parameter

#### BitField

RLE-coded bitfield of available functions.

Remark: FktIDs are 12-bit encoded!

Basis data type	Length	Condition	Description
Stream		-	<FktID1>{, <FktID2>}...

## 2.1.2 Notification (0x001)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support the notification mechanism. This property administrates the Notification Matrix of an FBlock.

### 2.1.2.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	Notification (0x001)	Set	<a href="#">Control</a> , <a href="#">DeviceID</a> , <a href="#">FktIDList</a>
		Get	<a href="#">FktID</a>
		Status	<a href="#">FktID</a> , <a href="#">DeviceIDList</a>
		Error	ErrorCode, ErrorInfo

### 2.1.2.2 Parameter

#### Control

The parameter Control determines where the entry has to be done or the deletion respectively.

- SetAll = Entry of DeviceID in all properties that support notification.
- SetFunction = Entry of DeviceID for the specified functions in the Notification Matrix.
- ClearAll = Deletion of DeviceID at all functions of the Notification Matrix.
- ClearFunction = Deletion of DeviceID for the specified functions in the Notification Matrix.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x03	0x00	SetAll	SetAll
		0x01	SetFunction	SetFunction
		0x02	ClearAll	ClearAll
		0x03	ClearFunction	ClearFunction

#### DeviceID

Either RxTxLog of a device or a group address.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktIDList

List of functions.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">FktID</a> (repeated)



## FktID

---

Function ID.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## DeviceIDList

---

List of devices.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">DeviceID</a> (repeated)

## 2.1.3 NotificationCheck (0x002)

Occurrence: Optional

Under certain system conditions, it can be helpful if a device can check whether its entries are still existent in the Notification Matrix. In case of an error, a device is able to renew its entries.

### 2.1.3.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	NotificationCheck (0x002)	Get	<a href="#">DeviceID</a>
		Status	<a href="#">DeviceID</a> , <a href="#">FktIDList</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.3.2 Parameter

#### DeviceID

Either RxTxLog of a device or a group address.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktIDList

List of functions.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">FktID</a> (repeated)

#### FktID

Function ID.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### ErrorCode

(Refer to the MOST Specification, section "Structure of MOST Messages: OPType")

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ErrorInfo

---

Besides the error information provided in the MOST Specification, additional function specific error info values are specified here.

Basis data type	Length	Condition	Description
Stream		ErrorCode = 0x20	Content: <a href="#">ErrorInfo_Detail</a>

## ErrorInfo\_Detail

---

Error info details.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x01...0x01	0x01	BufferOverflow	Buffer Overflow List of functions (parameter FktIDList of Status Message) is too long to be transmitted correctly by the message service, caused by a configuration fault.

## 2.1.4 Version (0x010)

Occurrence: Optional

This function is deprecated and only kept for compatibility reasons! It is superseded by FBlockInfo (0x011).

It describes the version of the FBlock, divided into the major version, the minor version and the build number. (E.g., Version 2.3.5)

### 2.1.4.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	Version (0x010)	Get	-
		Status	<a href="#">Major</a> , <a href="#">Minor</a> , <a href="#">Build</a>
		Error	ErrorCode, ErrorInfo

### 2.1.4.2 Parameter

#### Major

Major version value of the FBlock. A difference in the major version indicates that some changes are not backward compatible.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### Minor

Minor version value of the FBlock. A difference in the minor version indicates enhanced functionality.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### Build

Build number of the FBlock. A difference in the build number indicates typographical changes as well as bugfixes.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## 2.1.5 FBlockInfo (0x011)

Occurrence: Conditional

To distinguish between the functionality of similar FBlocks with the same FBlockID, the property FBlockInfo has to be implemented by every FBlock, except System FBlocks.

The function FBlockInfo provides information about the FBlock name, the name of the instance, the corresponding MOST Specification version, and the version of the FBlock itself. All this information is provided in text form.

The name of the instance is a string, which is assigned by the system integrator to allow a further differentiation of the application, e.g., "DISP\_L" for a left display or "DISP\_R" for a right display).

### 2.1.5.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	FBlockInfo (0x011)	Get	<a href="#">ID</a>
		Status	<a href="#">ID</a> , <a href="#">Description</a>
		Error	ErrorCode, ErrorInfo

### 2.1.5.2 Parameter

#### ID

The ID parameter corresponds to one entry of the Description parameter.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Description

Description provides either information about an entire FBlock or a particular function. Which is requested, depends on the parameter ID:

IDs equal to or larger than 0xF000 indicate that information about the FBlock is requested, for example, version of the FBlock.

If ID is in the range 0x0000...0x0FFF, ID corresponds to a function ID. In this case, information on the maturity of that particular function is requested (partly implemented, fully implemented...).

Basis data type	Length	Condition	Description
Stream		ID = 0x00 ... 0xFFFF	Content: <a href="#">FunctionMaturity</a>
		ID = 0xF000	Content: <a href="#">FBlockName</a> , <a href="#">SupplierVersion</a> , <a href="#">FBlockVersion</a> , <a href="#">MOSTVersion</a> , <a href="#">SystemIntegrator</a> , <a href="#">FBlockType</a>
		ID = 0xF001	Content: <a href="#">FBlockName</a>
		ID = 0xF002	Content: <a href="#">SupplierVersion</a>
		ID = 0xF003	Content: <a href="#">FBlockVersion</a>
		ID = 0xF004	Content: <a href="#">MOSTVersion</a>
		ID = 0xF005	Content: <a href="#">SystemIntegrator</a>
		ID = 0xF006	Content: <a href="#">FBlockType</a>

## FunctionMaturity

FunctionMaturity defines the implementation progress of one specific function that is identified by parameter ID.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x12	0x00	Unknown	Unknown
		0x01	InterfaceOnly	Interface only
		0x02	PartlyImplemented	Partly implemented
		0x03	FullyImplemented	Fully implemented
		0x11	PartlyImplementedVerified	Partly implemented and verified
		0x12	FullyImplementedVerified	Fully implemented and verified

## FBlockName

Name of the FBlock corresponding to the FBlockID as specified in the FBlock library, e.g. "AmFmTuner".

Basis data type	MaxSize
String	#NULL#

## SupplierVersion

String, of format "x.y.z", where x,y,z are one or two digit numbers, which describe the version of the FBlock from the system integrator function catalog.

Basis data type	MaxSize
String	#NULL#

## FBlockVersion

This string has the format "x.y.z", where x,y, and z are one or two digit numbers. The combination x.y.z describes the FBlock version that the FBlock is based upon. In proprietary FBlocks this field contains "0.0.0".

Basis data type	MaxSize
String	#NULL#

## MOSTVersion

This string has the format "x.y.z", where x,y, and z are one or two digit numbers. The combination x.y.z describes the MOST Specification version that the FBlock is based upon. (This indicates, which version of the GeneralFBlock matches the implemented FBlock.)

Basis data type	MaxSize
String	#NULL#

## SystemIntegrator

---

String, which identifies the system integrator responsible for the system (e.g., "Audi", "BMW", "Daimler", or "Volvo"). A value "MOSTCo" specifies a generic FBlock implemented according to the MOST FBlock library.

Basis data type	MaxSize
String	#NULL#

## FBlockType

---

This string, which is assigned by the system integrator, allows a further differentiation of FBlock instances, e.g., "DISP\_L" for a left display or "DISP\_R" for a right display.

Basis data type	MaxSize
String	#NULL#

## 2.1.6 DynArrayIns (0x080)

Occurrence: Optional

Insert a number (Quantity) of new array rows after the line that is identified by the parameter Tag.

### 2.1.6.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DynArrayIns (0x080)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktID</a> , <a href="#">Tag</a> , <a href="#">Quantity</a> , <a href="#">InsertData</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a>

### 2.1.6.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktID

Function ID of the DynamicArray.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Tag

Unique handle of a row of an array (0xFFFF = no valid value).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Quantity

Number of rows.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none



## InsertData

---

Data to be inserted.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">Entry</a> (repeated)

## Entry

---

The contents of an entry, including its Tag as the first parameter.

Basis data type	Length	Description
Stream		

## 2.1.7 DynArrayDel (0x081)

Occurrence: Optional

Delete a number (Quantity) of array rows in a dynamic array after (inclusive) the row, which is identified by the parameterTag.

### 2.1.7.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DynArrayDel (0x081)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktID</a> , <a href="#">Tag</a> , <a href="#">Quantity</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a>

### 2.1.7.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktID

Function ID of the DynamicArray.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0	0x0...0xFFFF	1	none

#### Tag

Unique handle of a row of an array (0xFFFF = no valid value).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Quantity

Number of rows.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.8 MapIns (0x082)

Occurrence: Optional

Using the method MapIns, a number of array elements (entire lines) will be inserted in the Map with the given FktID. Because the elements of a Map are not ordered, no given position for the insertion can be specified. The number of array elements to insert is given in the parameter Quantity. The data contents of the lines to be inserted will be transferred in the parameter Data of type Stream. Because the Slave is responsible for assigning Tags, the Tag values in Data must be ignored (to avoid misunderstandings, these values should be set to 0xFFFF).

In case of using StartResult and if the insertion of the elements has been successful, the assigned Tags will be returned in the parameter TagList of the Result message. The items in TagList must have the same order as the items in the parameter Data of the corresponding StartResult request.

### 2.1.8.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	MapIns (0x082)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktID</a> , <a href="#">Quantity</a> , <a href="#">InsertData</a>
		AbortAck	<a href="#">SenderHandle</a>
		StartAck	<a href="#">SenderHandle</a> , <a href="#">FktID</a> , <a href="#">Quantity</a> , <a href="#">InsertData</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">Quantity</a> , <a href="#">TagList</a>

### 2.1.8.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktID

The function ID of the Map property, for which the insert operation is to be applied.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Quantity

The number of elements that are to be inserted into the Map property.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## InsertData

A list with the contents of the entries to insert into the specified datatype. The length of this parameter has to match the specified quantity and the format of the Map property. The values of the tags shall be set to 0xFFFF as they are assigned by the slave providing the Map property.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">Entry</a> (repeated)

## Entry

The contents of an entry of the Map, including its Tag as the first parameter.

Basis data type	Length	Description
Stream		

## TagList

A list with the tags that have been created for the inserted entries. The entries shall be in the same order as the entries specified in the parameter InsertData.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">Tag</a> (repeated)

## Tag

A tag that identifies an entry of the Map.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.9 MapDel (0x083)

Occurrence: Optional

The method MapDel deletes a number of array elements (entire lines) from the Map with the given FktID. The number of elements to delete is described through the parameter Quantity of data type Unsigned Word. The Tags of lines to be deleted are contained in the parameter TagList of type Stream. Because the elements in a Map are not ordered, no deletion of ranges is possible. However, using a Quantity of 0xFFFF and an empty TagList will delete the whole Map.

If the method MapDel is called with the OPType StartResultAck and if the deletion of the lines is successful, the given FktID is returned together with the number and the tags of the deleted lines in the parameters Quantity and TagList.

### 2.1.9.1 Format of Function

Function classes: Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	MapDel (0x083)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktID</a> , <a href="#">Quantity</a> , <a href="#">TagList</a>
		AbortAck	<a href="#">SenderHandle</a>
		StartAck	<a href="#">SenderHandle</a> , <a href="#">FktID</a> , <a href="#">Quantity</a> , <a href="#">TagList</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">FktID</a> , <a href="#">Quantity</a> , <a href="#">TagList</a>

### 2.1.9.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktID

The function ID of the Map property, for which the delete operation is to be applied.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Quantity

Describes the number of elements that are to be deleted from the Map property. If this parameter has the value 0xFFFF, all entries of the Map property are to be deleted.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## TagList

---

A list with the tags that have to be deleted from the Map property (if parameter Quantity is not 0xFFFF).

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">Tag</a> (repeated)

## Tag

---

A tag that identifies an entry of the Map.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.10 CreateArrayWindow (0x090)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that implement a LongArray. Create a new instance of an ArrayWindow.

### 2.1.10.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	CreateArrayWindow (0x090)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktIDMotherArray</a> , <a href="#">Tag</a> , <a href="#">WindowSize</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">FktIDArrayWindow</a>

### 2.1.10.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktIDMotherArray

FktID of the MotherArray.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Tag

Unique handle of a row of an array (0xFFFF = no valid value).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### WindowSize

Size of the ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## FktIDArrayWindow

---

Function ID of the created ArrayWindow.

**Note:** The valid range of *FktIDArrayWindow* depends on the *FktID* range reserved in the description of the respective *LongArray*.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none



## 2.1.11 DestroyArrayWindow (0x091)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that implement a LongArray.  
Delete an ArrayWindow.

### 2.1.11.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DestroyArrayWindow (0x091)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktIDArrayWindow</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a>

### 2.1.11.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktIDArrayWindow

Function ID of the created ArrayWindow.

**Note:** The valid range of FktIDArrayWindow depends on the FktID range reserved in the description of the respective LongArray.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.12 MoveArrayWindow (0x092)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that implement a LongArray. This function is used to position an ArrayWindow on a MotherArray.

### 2.1.12.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	MoveArrayWindow (0x092)	StartAck	<a href="#">SenderHandle</a> , <a href="#">FktIDArrayWindow</a> , <a href="#">MovingMode</a> , <a href="#">Number</a> , <a href="#">Tag</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.12.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktIDArrayWindow

Function ID of the ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### MovingMode

Determines the desired positioning mode for the ArrayWindow.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x04	0x00	Top	Top
		0x01	Bottom	Bottom
		0x02	Up	Up
		0x03	Down	Down
		0x04	Absolute	Absolute

#### Number

Number of lines to move the ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## Tag

---

Unique handle of a row of an array (0xFFFF = no valid value).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.13 SearchArrayWindow (0x093)

Occurrence: Optional

The property is for searching the MotherArray for a text (SearchString) in the column indicated by PosY. If the text could not be found, the device answers with the ErrorCode 0x07 (parameter not available). A detailed description of this method can be found in the MOST specification.

### 2.1.13.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	SearchArrayWindow (0x093)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktIDArrayWindow</a> , <a href="#">PosY</a> , <a href="#">SearchString</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a>

### 2.1.13.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktIDArrayWindow

Function ID of the ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### PosY

PosY indicates which column is the target of the search.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### SearchString

The text to search for.

Basis data type	MaxSize
String	100

## 2.1.14 LongArrayInfo (0x094)

Occurrence: Optional

Each Slave with one or several LongArrays offers the property LongArrayInfo (one instance for all LongArrays). It enables Controllers to re-synchronize after a system error or interrupt. Controllers can check whether ArrayWindows, previously created in the Slave, still exist.

### 2.1.14.1 Format of Function

**Function classes:** Array of { Record of { Number Number Number } }

FBlock	Function	OPType	Parameter
GeneralFBlock	LongArrayInfo (0x094)	Get	<a href="#">Pos</a>
		Status	<a href="#">Pos</a> , <a href="#">Data</a>
		Error	ErrorCode, ErrorInfo

### 2.1.14.2 Parameter

#### Pos

The parameter Pos={x,y} consists of two bytes, x and y, and shows which parameter shall be set or read.

Valid range: x=0..NMax, y=0..3

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Data

This Array contains individual records with information about present LongArrays, existing ArrayWindows, and the Controllers that requested their creation.

Basis data type	Length	Description	
Array	-	Pos	Data
		{ x=0, y=0 }	{FktIDMA[1], FktIDAW[1], DeviceIDController[1],...,FktIDMA[NMax], FktIDAW[NMax], DeviceIDController[NMax]}
		{ x>0, y=0 }	{FktIDMA[x], FktIDAW[x], DeviceIDController[x]}
		{ x>0, y=1 }	{FktIDMA[x]}
		{ x>0, y=2 }	{FktIDAW[x]}
		{ x>0, y=3 }	{DeviceIDController[x]}

#### FktIDMA

Function ID of a MotherArray.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## FktIDAW

---

Function ID of an ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## DeviceIDController

---

DeviceID of the Controller that created the ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.15 ArrayWindowIns (0x09A)

Occurrence: Optional

Inserts a number of lines into a MotherArray.

### 2.1.15.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	ArrayWindowIns (0x09A)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktIDArrayWindow</a> , <a href="#">Tag</a> , <a href="#">Quantity</a> , <a href="#">InsertData</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">TagList</a>

### 2.1.15.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktIDArrayWindow

Function ID of the ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Tag

Unique handle of a row (0xFFFF no valid value) after which the new lines are inserted.  
0x0000 indicates an insertion before the first line of the ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Quantity

Number of rows.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## InsertData

---

Data to be inserted.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">Entry</a> (repeated)

## Entry

---

The contents of an entry. Tag is not included but is determined by the slave.

Basis data type	Length	Description
Stream		

## TagList

---

The list of tags that were generated by the ArrayWindowIns function. The order of the tags corresponds to the order in which the entries were received.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">Tag</a> (repeated)



## 2.1.16 ArrayWindowDel (0x09B)

Occurrence: Optional

Deletes a number of lines from a MotherArray.

### 2.1.16.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	ArrayWindowDel (0x09B)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FktIDArrayWindow</a> , <a href="#">Tag</a> , <a href="#">Quantity</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a>

### 2.1.16.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FktIDArrayWindow

Function ID of the ArrayWindow.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Tag

Unique handle of a row (0xFFFF no valid value) after which the given number of lines is deleted.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Quantity

Number of rows.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.17 PowerDownDelay (0x0A0)

Occurrence: Optional

Some devices need to stay powered on after shutdown of the MOST network for a configurable time to provide internal functions.

If parameter Duration is set to 0, no Powerdown Delay is active.

**Note:** The timer *tPowerSwitchOffDelay*, specifying the minimum time of five seconds, still applies.

If Duration is set to 5, the respective ECU stays on 5 minutes after shutdown of MOST.

The duration starts with a Signal Off event preceded by NetBlock.Shutdown.Start(Execute).

### 2.1.17.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	PowerDownDelay (0x0A0)	Set	<a href="#">Duration</a>
		Get	-
		SetGet	<a href="#">Duration</a>
		Status	<a href="#">Duration</a>
		Error	ErrorCode, ErrorInfo

### 2.1.17.2 Parameter

#### Duration

Time to stay powered-on after shutdown of MOST (Shutdown Execute).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	min

## 2.1.18 HDCP\_ReceiverConnectedIndication (0x0C0)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that assemble an HDCP transmitter.

### 2.1.18.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	HDCP_ReceiverCo nnectedIndication (0x0C0)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a>
		AbortAck	<a href="#">SenderHandle</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.18.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FBlockID

ID of the FBlock that owns the receiver.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### InstID

Instance ID of the FBlock instance that owns the receiver.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### ErrorCode

(Refer to the MOST Specification, section "Structure of MOST Messages: OPType")

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ErrorInfo

---

Besides the error information provided in the MOST Specification, additional function specific error info values are specified here.

Basis data type	Length	Condition	Description
Stream		ErrorCode = 0x20	Content: <a href="#">ErrorInfo_RecConnectedInd</a>

## ErrorInfo\_RecConnectedInd

---

Error info details.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x01...0x03	0x01	TooManyDevices	Too many devices
		0x02	TooManyLevels	Too many levels
		0x03	AuthenticationFailure	Authentication failure

## 2.1.19 HDCP\_ReceiverDisconnectedIndication (0x0C1)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that assemble an HDCP transmitter.

### 2.1.19.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	HDCP_ReceiverDisconnectedIndication (0x0C1)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a>
		AbortAck	<a href="#">SenderHandle</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.19.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FBlockID

ID of the FBlock that owns the receiver.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### InstID

Instance ID of the FBlock instance that owns the receiver.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### ErrorCode

(Refer to the MOST Specification, section "Structure of MOST Messages: OPType")

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ErrorInfo

---

Besides the error information provided in the MOST Specification, additional function specific error info values are specified here.

Basis data type	Length	Condition	Description
Stream		ErrorCode = 0x20	Content: <a href="#">ErrorInfo_RecDisconInd</a>

## ErrorInfo\_RecDisconInd

---

Error info details.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x01...0x01	0x01	DisconnectionFailure	Disconnection failure

## 2.1.20 HDCP\_Control (0x0C2)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support HDCP.

### 2.1.20.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	HDCP_Control (0x0C2)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">RequesterFBlockID</a> , <a href="#">RequesterInstID</a> , <a href="#">msg_id</a> , <a href="#">Control_HDCP</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a>
		Error	ErrorCode, ErrorInfo

### 2.1.20.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### RequesterFBlockID

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### RequesterInstID

Instance ID of the FBlock instance that owns the source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### msg\_id

HDCP message ID field.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x01.... 0x1F	0x01	NullMessage	Null message
		0x02	AKE_Init	AKE_Init (Transmitter to Receiver)
		0x03	AKE_Send_Cert	AKE_Send_Cert (Receiver to Transmitter)
		0x04	AKE_No_Stored_km	AKE_No_Stored_km

Basis data type	Range of values	Code	Symbolic Name	Description
				(Transmitter to Receiver)
		0x05	AKE_Stored_km	AKE_Stored_km (Transmitter to Receiver)
		0x06	AKE_Send_rrx	AKE_Send_rrx (Receiver to Transmitter)
		0x07	AKE_Send_H_prime	AKE_Send_H_prime (Receiver to Transmitter)
		0x08	AKE_Send_Pairing_Info	AKE_Send_Pairing_Info (Receiver to Transmitter)
		0x09	LC_Init	LC_Init (Transmitter to Receiver)
		0x0A	LC_Send_L_prime	LC_Send_L_prime (Receiver to Transmitter)
		0x0B	SKE_Send_Eks	SKE_Send_Eks (Transmitter to Receiver)
		0x0C	RepeaterAuth_Send_ReclD_List	RepeaterAuth_Send_ReceiverID_List (Receiver to Transmitter)
		0x0D	RTT_Ready	RTT_Ready (Receiver to Transmitter)
		0x0E	RTT_Challenge	RTT_Challenge (Transmitter to Receiver)
		0x0F	RepeaterAuth_Send_Ack	RepeaterAuth_Send_Ack (Transmitter to Receiver)
		0x10	RepeaterAuth_Stream_Manage	RepeaterAuth_Stream_Manage (Transmitter to Receiver)
		0x11	RepeaterAuth_Stream_Ready	RepeaterAuth_Stream_Ready (Receiver to Transmitter)
		0x12	Receiver_AuthStatus	Receiver_AuthStatus (Receiver to Transmitter)
		0x13	AKE_Transmitter_Info	AKE_Transmitter_Info (Transmitter to Receiver)
		0x14	AKE_Receiver_Info	AKE_Receiver_Info (Receiver to Transmitter)
		0x15 ... 0x1F		Reserved

## Control\_HDCP

HDCP control message envelope.

Basis data type	Length	Description
Stream		



## 2.1.21 HDCP\_DecipherStatus (0x0C4)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that assemble an HDCP receiver. This function provides information about the decipher status.

### 2.1.21.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	HDCP_DecipherStatus (0x0C4)	Get	<a href="#">SinkNr</a>
		Status	<a href="#">SinkNr</a> , <a href="#">DecipherStatus</a>
		Error	ErrorCode, ErrorInfo

### 2.1.21.2 Parameter

#### SinkNr

Number of a data sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### DecipherStatus

CipherError provides the current state of the deciphering machine.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x01	0x00	NoError	No error
		0x01	DecodingError	Decoding error

## 2.1.22 HDCP\_Assign (0x0C5)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support HDCP with several HDCP transmitters.

This functions makes the assignment of the HDCP transmitter to the sink

**Note:** Function Connect does not provide this information as it is based on the connection label only.

### 2.1.22.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	HDCP_Assign (0x0C5)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">SinkNr</a> , <a href="#">HDCP_Transmitter_FBlockID</a> , <a href="#">HDCP_Transmitter_InstID</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a>
		Error	ErrorCode, ErrorInfo

### 2.1.22.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### SinkNr

Number of a data sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### HDCP\_Transmitter\_FBlockID

ID of the transmitter FBlock.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### HDCP\_Transmitter\_InstID

ID of the transmitter FBlock instance.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## 2.1.23 SourceInfo (0x100)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that contain a source. This property gives particulars about the type of streaming data a source can handle. If notification is used for an FBlock with multiple sources, multiple notifications will be sent when the status changes for more than one source.

For further information please refer to the MOST Stream Transmission Specification.

### 2.1.23.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceInfo (0x100)	Get	<a href="#">SourceNr</a>
		Status	<a href="#">SourceNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a> , <a href="#">TransmissionClass</a> , <a href="#">ContentProtection</a> , <a href="#">ContentType</a> , <a href="#">ContentDescription</a> , <a href="#">TransmissionParameter</a>
		Error	ErrorCode, ErrorInfo

### 2.1.23.2 Parameter

#### SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### BlockWidth

Number of transferred bytes per MOST frame.

Speed grade	Parameter value range
MOST50	1...117
MOST150	1...372

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	Byte

#### ConnectionLabel

Connection identifier. If no bandwidth was allocated yet, 0x0000 must be used.

Speed grade	Parameter value range
MOST50	0; 11...127

Speed grade	Parameter value range
MOST150	0; 12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## TransmissionClass

Describe the used MOST transmission type.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x04	0x00	Synchronous	Synchronous
		0x01	DiscreteFrameIsochronous	DiscreteFrame Isochronous
		0x02	PacketizedIsochronous	Packetized Isochronous
		0x03	QoSIP	QoS IP
		0x04	Asynchronous	Asynchronous

## ContentProtection

Type of used Content Protection.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x05	0x00	None	None
		0x01	SerialCopyManagement	Serial Copy Management
		0x02	MOSTDTCP	MOST DTCP (Supplement B)
		0x03	DTCPIP	DTCP-IP (Supplement E); deprecated!
		0x04	MOSTDTCP_SuppH	MOST DTCP (Supplement H)
		0x05	HDCP	HDCP

## ContentType

Type of transported content.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00.... 0xFE	0x00	Audio	Audio
		0x01	ReservedDeprecatedCDROM	Reserved, usage deprecated (CDROM)
		0x02	SPDIF	SPDIF
		0x08	SAD	SAD
		0x0F	PhaseInformation	Phase Information
		0x10	GenericPCM	GenericPCM
		0x11	ReservedDeprecatedPCMDTCP	Reserved, usage deprecated (GenericPCM/DTCP)
		0x12	GenericPCM_w_FRM	GenericPCM_with_FrameRateMultiplier
		0x20	MPEG1SystemStream	MPEG1_SystemStream
		0x21	MPEG2ProgramStream	MPEG2_ProgramStream
		0x22	MPEG2TransportStream	MPEG2_TransportStream
		0x30 ...		Reserved for MPEG

Basis data type	Range of values	Code	Symbolic Name	Description
		0x3F		Elementary Streams
		0x40	ReservedDeprecatedMPEG1	Reserved, usage deprecated (MPEG1 DTCP System Stream)
		0x41	ReservedDeprecatedMPEG2Pr	Reserved, usage deprecated (MPEG2 DTCP Program Stream)
		0x42	ReservedDeprecatedMPEG2Tr	Reserved, usage deprecated (MPEG2 DTCP Transport Stream)
		0x50 ... 0x5F		Reserved for Compressed Audio
		0x90	IPStream	IP Stream (deprecated)
		0x91	Ethernet	Ethernet
		0xC0 ... 0xEF		System Integrator specific
		0xF0 ... 0xFE		Supplier specific

## ContentDescription

The parameter depends on the used content type.

Basis data type	Max. Length	Condition	Description
Short Stream		ContentType = 0x00	Content: <a href="#">AudioChannels</a> , <a href="#">Resolution</a>
		ContentType = 0x02	Content: <a href="#">SpeedFactor</a>
		ContentType = 0x08	Content: <a href="#">DataConnectionLabel</a>
		ContentType = 0x0F	
		ContentType = 0x10	Content: <a href="#">AudioChannels</a> , <a href="#">AudioChannelList</a>
		ContentType = 0x12	Content: <a href="#">AudioChannels</a> , <a href="#">AudioChannelList_FRM</a>
		ContentType = 0x20	
		ContentType = 0x21	
		ContentType = 0x22	
		ContentType = 0x90	Content: <a href="#">IP Address</a> , <a href="#">Port</a>
		ContentType = 0x91	

## AudioChannels

Number of audio channels.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## Resolution

Resolution of the audio samples in bytes.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	Byte

## SpeedFactor

SpeedFactor defines the speed factor of the content with respect to the transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## DataConnectionLabel

Connection identifier of the SAD associated stream. If no bandwidth was allocated yet, 0x0000 must be used.

Speed grade	Parameter value range
MOST50	0; 11...127
MOST150	0; 12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## AudioChannelList

The parameter depends on the used transmission class.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">AudioChannelName</a> (repeated), <a href="#">BitsPerSample</a> (repeated)

## AudioChannelName

Describe the channel name and determine the room position of the channel.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x14	0x00	CurrentlyNotInUse	Currently not in use
		0x01	FrontLeft	Front Left (FL)
		0x02	FrontRight	Front Right (FR)
		0x03	FrontCenter	Front Center (FC)
		0x04	LowFrequency	Low Frequency (LF)
		0x05	BackLeft	Back Left (BL)
		0x06	BackRight	Back Right (BR)
		0x07	FrontLeftCenter	Front Left of Center (FLC)
		0x08	FrontRightCenter	Front Right of Center (FRC)
		0x09	BackCenter	Back Center (BC)
		0x0A	SideLeft	Side Left (SL)
		0x0B	SideRight	Side Right (SR)
		0x0C	TopCenter	Top Center (TC)
		0x0D	TopFrontLeft	Top Front Left (TFL)
		0x0E	TopFrontCenter	Top Front Center (TFC)
		0x0F	TopFrontRight	Top Front Right (TFR)
		0x10	TopBackLeft	Top Back Left (TBL)
		0x11	TopBackCenter	Top Back Center (TBC)
		0x12	TopBackRight	Top Back Right (TBR)
		0x13	BackLeftCenter	Back Left of Center (BCL)
		0x14	BackRightCenter	Back Right of Center

Basis data type	Range of values	Code	Symbolic Name	Description
				(BCR)

## BitsPerSample

Resolution of the audio samples in bits.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	bit

## AudioChannelList\_FRM

The parameter depends on the used transmission class.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">AudioChannelName</a> (repeated), <a href="#">BitsPerSample</a> (repeated), <a href="#">FrameRateMultiplier</a> (repeated)

## FrameRateMultiplier

Specifies the multiplier of the audio frame rate compared to MOST frame rate (1,2 or 4).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte		1...4	1	none

## IP\_Address

ContentType "IP Stream" (0x90) is deprecated!

Length parameter decides if IP\_V4 (32bit address) or IP\_V6 (128bit) address is used.

- Length=4 means IP\_V4
- Length=16 means IP\_V6

Basis data type	Max. Length	Description
Short Stream		

## Port

ContentType "IP Stream" (0x90) is deprecated!

16 bit Port.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## TransmissionParameter

The parameter depends on the used transmission class.

Basis data type	Length	Condition	Description
Stream		TransmissionClass = 0x00	-
		TransmissionClass = 0x01	Content: <a href="#">DataFrameBlockwidth</a> , <a href="#">DataSampleFrequency</a> , <a href="#">ClkSrcLabel</a>
		TransmissionClass = 0x02	Content: <a href="#">DataRate</a>
		TransmissionClass = 0x03	Content: <a href="#">DataRate</a>
		TransmissionClass = 0x04	-

## DataFrameBlockwidth

Number of bytes the original data frame uses.

**Note:** *DataFrameBlockwidth* is set to 0x00 for *ContentType* 0x0F (Phase Information).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	Byte

## DataSampleFrequency

Sample frequency of the original data frame in Hz.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Long	0		1	Hz

## ClkSrcLabel

Connection label for Phase Information. If no bandwidth was allocated yet, 0x0000 must be used.

Speed grade	Parameter value range
MOST50	0; 11...127
MOST150	0; 12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## DataRate

The maximum data rate of the original stream in kbit/s.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Long	0		1	kbps



## 2.1.24 Allocate (0x101)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that contain a source. The Allocate method causes the source to occupy bandwidth for streaming data.

### 2.1.24.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	Allocate (0x101)	StartResult	<a href="#">SourceNr</a>
		StartResultAck	<a href="#">SenderHandle</a> , <a href="#">SourceNr</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	<a href="#">SourceNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">SourceNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.24.2 Parameter

#### SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### ErrorCode

(Refer to the MOST Specification, section "Structure of MOST Messages: OPType")

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ErrorInfo

Besides the error information provided in the MOST Specification, additional function specific error info values are specified here.

Basis data type	Length	Condition	Description
Stream		ErrorCode = 0x20	Content: <a href="#">ErrorInfo_SourceNr</a> , <a href="#">ErrorInfo_BlockWidth</a>

## ErrorInfo\_SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## ErrorInfo\_BlockWidth

The requested BlockWidth.

Speed grade	Parameter value range
MOST50	1...117
MOST150	1...372

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	Byte

## BlockWidth

Number of transferred bytes per MOST frame.

Speed grade	Parameter value range
MOST50	1...117
MOST150	1...372

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	Byte

## ConnectionLabel

Connection identifier.

Speed grade	Parameter value range
MOST50	11...127
MOST150	12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.25 DeAllocate (0x102)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that contain a source. The method DeAllocate causes the source to free occupied Streaming bandwidth.

### 2.1.25.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DeAllocate (0x102)	StartResult	<a href="#">SourceNr</a>
		StartResultAck	<a href="#">SenderHandle</a> , <a href="#">SourceNr</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	<a href="#">SourceNr</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">SourceNr</a>
		Error	ErrorCode, ErrorInfo

### 2.1.25.2 Parameter

#### SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.26 SourceActivity (0x103)

Occurrence: Optional

This method controls the activity of a source.

### 2.1.26.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceActivity (0x103)	StartResult	<a href="#">SourceNr, Activity</a>
		StartResultAck	<a href="#">SenderHandle, SourceNr, Activity</a>
		ErrorAck	<a href="#">SenderHandle, ErrorCode, ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	<a href="#">SourceNr, Activity</a>
		ResultAck	<a href="#">SenderHandle, SourceNr, Activity</a>
		Error	ErrorCode, ErrorInfo

### 2.1.26.2 Parameter

#### SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### Activity

Through parameter Activity, streaming data transfer can be started, stopped, or paused.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x02	0x00	Off	Off
		0x01	Pause	Pause
		0x02	On	On

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.27 SourceName (0x104)

Occurrence: Optional

By property SourceName, an identifier of the streaming source data can be requested. If notification is used for an FBlock with multiple sources, multiple notifications will be sent when the status changes for more than one source.

### 2.1.27.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceName (0x104)	Get	<a href="#">SourceNr</a>
		Status	<a href="#">SourceNr</a> , <a href="#">SourceName</a>
		Error	ErrorCode, ErrorInfo

### 2.1.27.2 Parameter

#### SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### SourceName

Name of the source.

Basis data type	MaxSize
String	#NULL#

## 2.1.28 AllocateExt (0x108)

Occurrence: Optional

The AllocateExt method causes the source to occupy bandwidth for streaming data.

### 2.1.28.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	AllocateExt (0x108)	StartResult	<a href="#">SourceNr</a> , <a href="#">ClkSrcLabel</a>
		StartResultAck	<a href="#">SenderHandle</a> , <a href="#">SourceNr</a> , <a href="#">ClkSrcLabel</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	<a href="#">SourceNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a> , <a href="#">ClkSrcLabel</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">SourceNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a> , <a href="#">ClkSrcLabel</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.28.2 Parameter

#### SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### ClkSrcLabel

Connection label for Phase Information. The value 0xFFFF indicates that the clock channel is not used.

The value 0xFFFE instructs the source to allocate a new clock source.

Speed grade	Parameter value range
MOST50	11...127, 0xFFFE, 0xFFFF
MOST150	12...383, 0xFFFE, 0xFFFF

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## ErrorCode

(Refer to the MOST Specification, section "Structure of MOST Messages: OPType")

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ErrorInfo

Besides the error information provided in the MOST Specification, additional function specific error info values are specified here.

Basis data type	Length	Condition	Description
Stream		ErrorCode = 0x20	Content: <a href="#">ErrorInfo_SourceNr</a> , <a href="#">ErrorInfo_BlockWidth</a>

## ErrorInfo\_SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## ErrorInfo\_BlockWidth

The requested BlockWidth.

Speed grade	Parameter value range
MOST50	1...117
MOST150	1...372

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	Byte

## BlockWidth

Number of transferred bytes per MOST frame.

Speed grade	Parameter value range
MOST50	1...117
MOST150	1...372

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	Byte

## ConnectionLabel

Connection identifier.

Speed grade	Parameter value range
MOST50	11...127
MOST150	12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none



## 2.1.29 SinkInfo (0x110)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that contain a sink. This property gives particulars about the type of streaming sink data. If notification is used for an FBlock with multiple sinks, multiple notifications will be sent when the status changes for more than one sink. For further information please refer to the MOST Stream Transmission Specification.

### 2.1.29.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SinkInfo (0x110)	Get	<a href="#">SinkNr</a>
		Status	<a href="#">SinkNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a> , <a href="#">TransmissionClass</a> , <a href="#">ContentProtection</a> , <a href="#">ContentType</a> , <a href="#">ContentDescription</a> , <a href="#">TransmissionParameter</a>
		Error	ErrorCode, ErrorInfo

### 2.1.29.2 Parameter

#### SinkNr

Number of a data sink (within one FBlock there can be more than one), for example, 0x01 for the first sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### BlockWidth

Number of transferred bytes per MOST frame.

Speed grade	Parameter value range
MOST50	1...117
MOST150	1...372

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	Byte

#### ConnectionLabel

Connection identifier. If no bandwidth was allocated yet, 0x0000 must be used.

Speed grade	Parameter value range
MOST50	0; 11...127
MOST150	0; 12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## TransmissionClass

Describe the used MOST transmission type.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x04	0x00	Synchronous	Synchronous
		0x01	DiscreteFrameIsochronous	DiscreteFrame Isochronous
		0x02	PacketizedIsochronous	Packetized Isochronous
		0x03	QoSIP	QoS IP
		0x04	Asynchronous	Asynchronous

## ContentProtection

Type of used Content Protection.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x05	0x00	None	None
		0x01	SerialCopyManagement	Serial Copy Management
		0x02	MOSTDTCP	MOST DTCP (Supplement B)
		0x03	DTCP-IP	DTCP-IP (Supplement E); deprecated!
		0x04	MOSTDTCP_SuppH	MOST DTCP (Supplement H)
		0x05	HDCP	HDCP

## ContentType

Type of transported content.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00.... 0xFE	0x00	Audio	Audio
		0x01	ReservedDeprecatedCDROM	Reserved, usage deprecated (CDROM)
		0x02	SPDIF	SPDIF
		0x08	SAD	SAD
		0x0F	PhaseInformation	Phase Information
		0x10	GenericPCM	GenericPCM
		0x11	ReservedDeprecatedPCMDTCP	Reserved, usage deprecated (GenericPCM/DTCP)
		0x12	GenericPCM_w_FRM	GenericPCM_with_FrameRateMultiplier
		0x20	MPEG1SystemStream	MPEG1_SystemStream
		0x21	MPEG2ProgramStream	MPEG2_ProgramStream
		0x22	MPEG2TransportStream	MPEG2_TransportStream
		0x30 ... 0x3F		Reserved for MPEG Elementary Streams
		0x40	ReservedDeprecatedMPE	Reserved, usage

Basis data type	Range of values	Code	Symbolic Name	Description
			G1	deprecated (MPEG1 DTCP System Stream)
		0x41	ReservedDeprecatedMPEG2Pr	Reserved, usage deprecated (MPEG2 DTCP Program Stream)
		0x42	ReservedDeprecatedMPEG2Tr	Reserved, usage deprecated (MPEG2 DTCP Transport Stream)
		0x50 ... 0x5F		Reserved for Compressed Audio
		0x90	IPStream	IP Stream (deprecated)
		0x91	Ethernet	Ethernet
		0xC0 ... 0xEF		System Integrator specific
		0xF0 ... 0xFE		Supplier specific

## ContentDescription

The parameter depends on the used content type.

Basis data type	Max. Length	Condition	Description
Short Stream		ContentType = 0x00	Content: <a href="#">AudioChannels</a> , <a href="#">Resolution</a>
		ContentType = 0x02	Content: <a href="#">SpeedFactor</a>
		ContentType = 0x08	Content: <a href="#">DataConnectionLabel</a>
		ContentType = 0x0F	
		ContentType = 0x10	Content: <a href="#">AudioChannels</a> , <a href="#">AudioChannelList</a>
		ContentType = 0x12	Content: <a href="#">AudioChannels</a> , <a href="#">AudioChannelList_FRM</a>
		ContentType = 0x20	
		ContentType = 0x21	
		ContentType = 0x22	
		ContentType = 0x90	Content: <a href="#">IP_Address</a> , <a href="#">Port</a>
		ContentType = 0x91	

## AudioChannels

Number of audio channels.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## Resolution

Resolution of the audio samples in bytes.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	Byte

## SpeedFactor

SpeedFactor defines the speed factor of the content with respect to the transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## DataConnectionLabel

Connection identifier of the SAD associated stream. If no bandwidth was allocated yet, 0x0000 must be used.

Speed grade	Parameter value range
MOST50	0; 11...127
MOST150	0; 12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## AudioChannelList

The parameter depends on the used transmission class.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">AudioChannelName</a> (repeated), <a href="#">BitsPerSample</a> (repeated)

## AudioChannelName

Describe the channel name and determine the room position of the channel.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x14	0x00	CurrentlyNotInUse	Currently not in use
		0x01	FrontLeft	Front Left (FL)
		0x02	FrontRight	Front Right (FR)
		0x03	FrontCenter	Front Center (FC)
		0x04	LowFrequency	Low Frequency (LF)
		0x05	BackLeft	Back Left (BL)
		0x06	BackRight	Back Right (BR)
		0x07	FrontLeftCenter	Front Left of Center (FLC)
		0x08	FrontRightCenter	Front Right of Center (FRC)
		0x09	BackCenter	Back Center (BC)
		0x0A	SideLeft	Side Left (SL)
		0x0B	SideRight	Side Right (SR)
		0x0C	TopCenter	Top Center (TC)
		0x0D	TopFrontLeft	Top Front Left (TFL)
		0x0E	TopFrontCenter	Top Front Center (TFC)
		0x0F	TopFrontRight	Top Front Right (TFR)
		0x10	TopBackLeft	Top Back Left (TBL)
		0x11	TopBackCenter	Top Back Center (TBC)
		0x12	TopBackRight	Top Back Right (TBR)
		0x13	BackLeftCenter	Back Left of Center (BCL)
		0x14	BackRightCenter	Back Right of Center (BCR)

## BitsPerSample

Resolution of the audio samples in bits.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	bit

## AudioChannelList\_FRM

The parameter depends on the used transmission class.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">AudioChannelName</a> (repeated), <a href="#">BitsPerSample</a> (repeated), <a href="#">FrameRateMultiplier</a> (repeated)

## FrameRateMultiplier

Specifies the multiplier of the audio frame rate compared to MOST frame rate (1,2 or 4).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte		1...4	1	none

## IP\_Address

ContentType "IP Stream" (0x90) is deprecated!

Length parameter decides if IP\_V4 (32bit address) or IP\_V6 (128bit) address is used.

- Length=4 means IP\_V4
- Length=16 means IP\_V6

Basis data type	Max. Length	Description
Short Stream		

## Port

ContentType "IP Stream" (0x90) is deprecated!

16 bit Port.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## TransmissionParameter

The parameter depends on the used transmission class.

Basis data type	Length	Condition	Description
Stream		TransmissionClass = 0x00	-
		TransmissionClass = 0x01	Content: <a href="#">DataFrameBlockwidth</a> , <a href="#">DataSampleFrequency</a> , <a href="#">ClkSrcLabel</a>

Basis data type	Length	Condition	Description
		TransmissionClass = 0x02	Content: <a href="#">DataRate</a>
		TransmissionClass = 0x03	Content: <a href="#">DataRate</a>
		TransmissionClass = 0x04	-

## DataFrameBlockwidth

Number of bytes the original data frame uses.

**Note:** DataFrameBlockwidth is set to 0x00 for ContentType 0x0F (Phase Information).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	Byte

## DataSampleFrequency

Sample frequency of the original data frame in Hz.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Long	0		1	Hz

## ClkSrcLabel

Connection label for Phase Information. If no bandwidth was allocated yet, 0x0000 must be used.

Speed grade	Parameter value range
MOST50	0; 11...127
MOST150	0; 12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## DataRate

The maximum data rate of the original stream in kbit/s.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Long	0		1	kbits

## 2.1.30 Connect (0x111)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that contain a sink. By use of the method Connect, channels for data streaming will be connected.

### 2.1.30.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	Connect (0x111)	StartResult	<a href="#">SinkNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a>
		StartResultAck	<a href="#">SenderHandle</a> , <a href="#">SinkNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	<a href="#">SinkNr</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">SinkNr</a>
		Error	ErrorCode, ErrorInfo

### 2.1.30.2 Parameter

#### SinkNr

Number of a data sink (within one FBlock there can be more than one), for example, 0x01 for the first sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### BlockWidth

Number of transferred bytes per MOST frame.

Speed grade	Parameter value range
MOST50	1...117
MOST150	1...372

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	Byte

#### ConnectionLabel

Connection identifier.

Speed grade	Parameter value range
MOST50	11...127
MOST150	12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none



## 2.1.31 Disconnect (0x112)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that contain a sink. By use of the method Disconnect, connections for data streaming will be disconnected.

### 2.1.31.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	Disconnect (0x112)	StartResult	<a href="#">SinkNr</a>
		StartResultAck	<a href="#">SenderHandle</a> , <a href="#">SinkNr</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	<a href="#">SinkNr</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">SinkNr</a>
		Error	ErrorCode, ErrorInfo

### 2.1.31.2 Parameter

#### SinkNr

Number of a data sink (within one FBlock there can be more than one), for example, 0x01 for the first sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.32 Mute (0x113)

Occurrence: Optional

This property is for setting and reading the mute status. If notification is used for an FBlock with multiple sinks, multiple notifications will be sent when the status changes for more than one sink.

### 2.1.32.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	Mute (0x113)	Get	<a href="#">SinkNr</a>
		SetGet	<a href="#">SinkNr</a> , <a href="#">Status</a>
		Status	<a href="#">SinkNr</a> , <a href="#">Status</a>
		Error	ErrorCode, ErrorInfo

### 2.1.32.2 Parameter

#### SinkNr

Number of a data sink (within one FBlock there can be more than one), for example, 0x01 for the first sink.

Value 0x00 is used to mute / de-mute all sinks.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### Status

The Status parameter determines whether a sink is muted.

Basis data type	Bit #	Code	Description
Unsigned Byte	Bit 0	False	Mute Off
		True	Mute On
	Bit 1 ... 7	-	reserved

## 2.1.33 SinkName (0x114)

Occurrence: Optional

By using property SinkName, a name for the streaming data can be requested. If notification is used for an FBlock with multiple sinks, multiple notifications will be sent when the status changes for more than one sink.

### 2.1.33.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SinkName (0x114)	Get	<a href="#">SinkNr</a>
		Status	<a href="#">SinkNr</a> , <a href="#">SinkName</a>
		Error	ErrorCode, ErrorInfo

### 2.1.33.2 Parameter

#### SinkNr

Number of a data sink (within one FBlock there can be more than one), for example, 0x01 for the first sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### SinkName

Name of the streaming data sink.

Basis data type	MaxSize
String	#NULL#

## 2.1.34 ConnectTo (0x115)

Occurrence: Optional

By calling this method, the sink will be occasioned to connect itself to a certain source.

### 2.1.34.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	ConnectTo (0x115)	StartResult	<a href="#">FBlockID</a> , <a href="#">InstID</a> , <a href="#">SourceNr</a>
		StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a> , <a href="#">SourceNr</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	<a href="#">FBlockID</a> , <a href="#">InstID</a> , <a href="#">SourceNr</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a> , <a href="#">SourceNr</a>
		Error	ErrorCode, ErrorInfo

### 2.1.34.2 Parameter

#### FBlockID

ID of the FBlock that owns the source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### InstID

Instance ID of the FBlock instance that owns the source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### SourceNr

Number of the source that the sink is to be connected with.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.35 StreamDataInfo (0x116)

Occurrence: Conditional

The purpose of property StreamDataInfo is obtaining the list of source numbers and sink numbers the FBlock provides. This function is required for FBlocks that contain a source or sink.

### 2.1.35.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	StreamDataInfo (0x116)	Get	-
		Status	<a href="#">SourceNrList</a> , <a href="#">SinkNrList</a>
		Error	ErrorCode, ErrorInfo

### 2.1.35.2 Parameter

#### SourceNrList

List of the source numbers. The length field of the Short Stream corresponds to the amount of source numbers.

Basis data type	Max. Length	Condition	Description
Short Stream		-	Content: <a href="#">SourceNr</a> (repeated)

#### SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### SinkNrList

List of the sink numbers. The length field of the Short Stream corresponds to the amount of sink numbers.

Basis data type	Max. Length	Condition	Description
Short Stream		-	Content: <a href="#">SinkNr</a> (repeated)

#### SinkNr

Number of a data sink (within one FBlock there can be more than one), for example, 0x01 for the first sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## 2.1.36 SinkRouting (0x117)

Occurrence: Optional

This property describes the relation between the sink numbers of the FBlock and the physically existing streaming data sinks (e.g., the mixer inputs of an amplifier). Use this property to determine which sink numbers are mutually exclusive.

### 2.1.36.1 Format of Function

**Function classes:** Array of { Number }

FBlock	Function	OPType	Parameter
GeneralFBlock	SinkRouting (0x117)	Get	<a href="#">Pos</a>
		Status	<a href="#">Pos</a> , <a href="#">Data</a>
		Error	ErrorCode, ErrorInfo

### 2.1.36.2 Parameter

#### Pos

The parameter Pos={x,y} consists of two bytes, x and y, and shows which parameter shall be set or read. Since this property has only one dimension, y is unused.

Valid range: x=1...number of sources (as given in StreamDataInfo), y=0

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Data

The content depends on the parameter Pos.

Basis data type	Length	Description	
Array	-	Pos	Data
		{ x=0 }	{PhysicalSink[1],...,PhysicalSink[NMax]}
		{ x> 0 }	{PhysicalSink[x]}

#### PhysicalSink

Number to identify the physical sink this logical sink number is related to. The physical sink numbers are tested on equality by the ConnectionMaster.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## 2.1.37 ConnectExt (0x118)

Occurrence: Optional

By use of the method ConnectExt, channels for data streaming will be connected.

### 2.1.37.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	ConnectExt (0x118)	StartResult	<a href="#">SinkNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a> , <a href="#">ClkSrcLabel</a>
		StartResultAck	<a href="#">SenderHandle</a> , <a href="#">SinkNr</a> , <a href="#">BlockWidth</a> , <a href="#">ConnectionLabel</a> , <a href="#">ClkSrcLabel</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	<a href="#">SinkNr</a> , <a href="#">ClkSrcLabel</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">SinkNr</a> , <a href="#">ClkSrcLabel</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.37.2 Parameter

#### SinkNr

Number of a data sink (within one FBlock there can be more than one), for example, 0x01 for the first sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### BlockWidth

Number of transferred bytes per MOST frame.

Speed grade	Parameter value range
MOST50	1...117
MOST150	1...372

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	Byte

#### ConnectionLabel

Connection identifier.

Speed grade	Parameter value range
MOST50	11..127
MOST150	12...383

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## ClkSrcLabel

Connection label for Phase Information. The value 0xFFFF indicates that the clock channel is not used.

Speed grade	Parameter value range
MOST50	11...127, 0xFFFF
MOST150	12...383, 0xFFFF

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none



## 2.1.38 DTCP\_StartProcess (0x120)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support Digital Transmission Content Protection.

Important:

Function ID 0x120 has an alias 0x12E. It is not recommended to use the alias.

However, system integrators are allowed to use Function ID 0x12E instead of 0x120 for compatibility reasons.

### 2.1.38.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_StartProcess (0x120)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a>
		AbortAck	<a href="#">SenderHandle</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a>
		Error	ErrorCode, ErrorInfo

### 2.1.38.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FBlockID

ID of the FBlock that owns the source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### InstID

Instance ID of the FBlock instance that owns the source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## 2.1.39 DTCP\_Control (0x121)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support Digital Transmission Content Protection.

Important:

Function ID 0x121 has an alias 0x12F. It is not recommended to use the alias.

However, System Integrators are allowed to use Function ID 0x12F instead of 0x121 for compatibility reasons.

This function transmits DTCP control commands and corresponding responses. DTCP\_Control is used to exchange the messages required to implement the Authentication and Key Exchange protocols.

### 2.1.39.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_Control (0x121)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">RequesterFBlockID</a> , <a href="#">RequesterInstID</a> , <a href="#">ctype_response</a> , <a href="#">AKE_ID</a> , <a href="#">OP1_subfunction</a> , <a href="#">OP2_AKE_procedure</a> , <a href="#">OP3_exchange_key</a> , <a href="#">OP4_subfunction_dependent</a> , <a href="#">number_status</a> , <a href="#">blocks_remaining</a> , <a href="#">data</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">RequesterFBlockID</a> , <a href="#">RequesterInstID</a> , <a href="#">ctype_response</a> , <a href="#">AKE_ID</a> , <a href="#">OP1_subfunction</a> , <a href="#">OP2_AKE_procedure</a> , <a href="#">OP3_exchange_key</a> , <a href="#">OP4_subfunction_dependent</a> , <a href="#">number_status</a> , <a href="#">blocks_remaining</a> , <a href="#">data</a>
		Error	ErrorCode, ErrorInfo

### 2.1.39.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## RequesterFBlockID

FBlock ID of the FBlock that sends a DTCP command.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## RequesterInstID

Instance ID of the FBlock instance that sends a DTCP command.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ctype\_response

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x0C	0x00	CONTROL	ctype
		0x01	STATUS	ctype
		0x09	ACCEPTED	response
		0x0A	REJECTED	response
		0x0C	STABLE	response

## AKE\_ID

The AKE\_ID field specifies the format of the AKE\_ID dependent field.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x00	0x00	AKE_ID_0	AKE_ID = 0

## OP1\_subfunction

The subfunction field specifies the operation of control commands.  
The parameter "isochronous\_channel\_number" of the CONTENT\_KEY\_REQ command is used in MOST for indicating the SourceNr in the range of 1...63.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x01...0xC0	0x01	CHALLENGE	
		0x02	RESPONSE	
		0x03	EXCHANGE_KEY	
		0x04	SRM	
		0x80	CONTENT_KEY_REQ	
		0xC0	AKE_CANCEL	

## OP2\_AKE\_procedure

For MOST systems, only Full Authentication is used.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x04...0x04	0x04	full_auth	Full Authentication procedure

## OP3\_exchange\_key

Each bit of the exchange\_key field corresponds to one or multiple exchange keys.

Basis data type	Bit #	Code	Description
Unsigned Byte	Bit 0	False	-
		True	Exchange Key for M6 Copy-never content
	Bit 1	False	-
		True	Exchange Key for M6 Copy-one-generation content
	Bit 2	False	-
		True	Exchange Key for M6 No-more-copies content
	Bit 3	False	-
		True	Exchange Key for AES-128

## OP4\_subfunction\_dependent

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## number\_status

The status field is used to notify the device issuing the command of the reason when the command results in a REJECTED response.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x10	0x00	NoError	No error
		0x01	NoMoreAuthentication	No more authentication procedures currently supported
		0x02	NoIsochronousOutput	Not used for MOST systems
		0x03	NoPointToPointConnection	Not used for MOST systems
		0x04	DTCPUnavailable	
		0x05	NoAC	Not used for MOST systems
		0x07	AnyOtherError	
		0x08	IncorrectCommandOrder	Only for testing
		0x09	AuthenticationFailed	Only for testing
		0x0F	NoInformation	
		0x10	DatafieldSyntaxError	Only for testing

## blocks\_remaining

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## data

Basis data type	Length	Description
Stream		

## 2.1.40 DTCP\_Status (0x122)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support Digital Transmission Content Protection.

This function transmits DTCP status commands and corresponding responses.

A device issuing a command can determine the size of data field that the target device can accept using the AKE status command.

### 2.1.40.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_Status (0x122)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">RequesterFBlockID</a> , <a href="#">RequesterInstID</a> , <a href="#">ctype_response</a> , <a href="#">AKE_ID</a> , <a href="#">OP1_subfunction</a> , <a href="#">OP2_AKE_procedure</a> , <a href="#">OP3_exchange_key</a> , <a href="#">OP4_subfunction_dependent</a> , <a href="#">status</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">RequesterFBlockID</a> , <a href="#">RequesterInstID</a> , <a href="#">ctype_response</a> , <a href="#">AKE_ID</a> , <a href="#">OP1_subfunction</a> , <a href="#">OP2_AKE_procedure</a> , <a href="#">OP3_exchange_key</a> , <a href="#">OP4_subfunction_dependent</a> , <a href="#">status</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.40.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### RequesterFBlockID

FBlock ID of the FBlock that sends a DTCP command.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### RequesterInstID

Instance ID of the FBlock instance that sends a DTCP command.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ctype\_response

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x0C	0x00	CONTROL	ctype
		0x01	STATUS	ctype
		0x09	ACCEPTED	response
		0x0A	REJECTED	response
		0x0C	STABLE	response

## AKE\_ID

The AKE\_ID field specifies the format of the AKE\_ID dependent field.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x00	0x00	AKE_ID_0	AKE_ID = 0

## OP1\_subfunction

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x01...0xC0	0x01	CHALLENGE	
		0x02	RESPONSE	
		0x03	EXCHANGE_KEY	
		0x04	SRM	
		0x80	CONTENT_KEY_REQ	
		0xC0	AKE_CANCEL	

## OP2\_AKE\_procedure

The initiator of the STATUS command sets this value initially to 0xFF to request the supported AKE procedures of the target device. For the reply, the target device overwrites this value with the supported AKE procedures.

For MOST systems, only Full Authentication is used.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x04...0xFF	0x04	full_auth	Full Authentication procedure
		0xFF	initial_value	Initial value used by the sender

## OP3\_exchange\_key

The initiator of the STATUS command sets this value initially to 0xFF (RequestKx) to request the supported exchange keys of the target device. For the reply, the target device overwrites this value with the supported exchange keys.

Basis data type	Bit #	Code	Description
Unsigned Byte	Bit 0	False	-
		True	Exchange Key for M6 Copy-never content
	Bit 1	False	-
		True	Exchange Key for M6 Copy-one-generation content
	Bit 2	False	-
		True	Exchange Key for M6 No-more-copies content
	Bit 3	False	-
		True	Exchange Key for AES-128

## OP4\_subfunction\_dependent

For the STATUS command, this value is not used and defined to be 0xFF.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0xFF...0xFF	0xFF	fixed_value	Fixed value

## status

The initiator of the STATUS command sets this value initially to 0xFF to request the current situation of the target device. For the reply, the target device overwrites this value with its current situation.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0xF0...0xFF	0xF0	NoError	
		0xF1	NoMoreAuthentication	
		0xF2	NoIsochronousOutput	Not used for MOST systems
		0xF3	NoPointToPointConnection	Not used for MOST systems
		0xF4	DTCPUnavailable	
		0xF7	AnyOtherError	
		0xF9	AuthenticationFailed	Only for testing
		0xFF	NoInformation	Issued by the initiator of the command, not recommended as response

## 2.1.41 DTCP\_CipherStatus (0x123)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support Digital Transmission Content Protection.

Important:

Function ID 0x123 has an alias 0x12D. It is not recommended to use the alias.

However, system integrators are allowed to use Function ID 0x12D instead of 0x123 for compatibility reasons.

This function gives information about the state of the AKE and Ciphering components.

Because the SourceSinkNr parameter does not distinguish between source or sink, the sink and source devices in a system have to be designed so that it is possible to distinguish between different sources and sinks (e.g., using different FBlocks or device unique source and sink numbers).

### 2.1.41.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_CipherStatus (0x123)	Get	<a href="#">SourceSinkNr</a>
		Status	<a href="#">SourceSinkNr</a> , <a href="#">AuthenticationState</a> , <a href="#">AvailableExchangeKeys</a> , <a href="#">CipherError</a>
		Error	ErrorCode, ErrorInfo

### 2.1.41.2 Parameter

#### SourceSinkNr

Number of a data source or sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### AuthenticationState

AuthenticationState gives the current state of Authentication as defined in Chapter 3 of the 5C DTCP Specification

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x05	0x00	A0Unauthenticated	State A0: Unauthenticated
		0x01	A1Authentication	State A1: Full Authentication
		0x02	A2RestrictedAutentication	State A2: Restricted Authentication
		0x03	A3Autenticated	State A3: Authenticated
		0x04	A4SendContentChannelKey	State A4: Send Content Channel Key
		0x05	A5InitializeDevice	State A5: Initialize Device



## AvailableExchangeKeys

AvailableExchangeKeys gives the current set of available ExchangeKeys.

Basis data type	Bit #	Code	Description
Unsigned Byte	Bit 0	False	-
		True	Exchange Key for M6 Copy-never content
	Bit 1	False	-
		True	Exchange Key for M6 Copy-one-generation content
	Bit 2	False	-
		True	Exchange Key for M6 No-more-copies content
	Bit 3	False	-
		True	Exchange Key for AES-128

## CipherError

CipherError gives the current state of the ciphering machines

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x20	0x00	NoError	No error
		0x10	EncodingError	Encoding Error
		0x20	DecodingError	Decoding Error

## 2.1.42 DTCP\_Info (0x124)

Occurrence: Optional

This function is deprecated. This function gives information about MOST DTCP parameters.

### 2.1.42.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_Info (0x124)	Get	<a href="#">SourceSinkNr</a>
		Status	<a href="#">SourceSinkNr</a> , <a href="#">PacketFormat</a> , <a href="#">MediaType</a> , <a href="#">Packetlength</a> , <a href="#">EncryptionFrameSize</a>
		Error	ErrorCode, ErrorInfo

### 2.1.42.2 Parameter

#### SourceSinkNr

Number of a data source or sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### PacketFormat

PacketFormat gives the packet format which is used by the source / sink (please see MOST Content Protection Scheme--DTCP Implementation).

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x04	0x00	NotDefined	Not defined
		0x01	NotApplicable	Not applicable
		0x02	GenericMOSTDTCP	Generic MOST-DTCP Format
		0x03	AVPacketizedMOSTDTC-P	A/V Packetized MOST-DTCP Format
		0x04	AVPacketizedDTCP-IP	A/V Packetized DTCP-IP; deprecated!

#### MediaType

This parameter refers to the MediaType values, which are given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## Packetlength

---

This parameter refers to the MOST Packet Length value, which is given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## EncryptionFrameSize

---

This parameter refers to the DTCP Encryption Frame Size value, which is given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.43 DTCP\_ContentKeyProcess (0x125)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support Digital Transmission Content Protection. The method starts the establishing of Content Keys.

### 2.1.43.1 Format of Function

**Function classes:** Sequence Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_ContentKey Process (0x125)	StartResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a> , <a href="#">SourceNr</a> , <a href="#">SinkNr</a> , <a href="#">PacketFormat</a> , <a href="#">MediaType</a> , <a href="#">Packetlength</a> , <a href="#">EncryptionFrameSize</a>
		ErrorAck	<a href="#">SenderHandle</a> , ErrorCode, ErrorInfo
		ProcessingAck	<a href="#">SenderHandle</a>
		ResultAck	<a href="#">SenderHandle</a> , <a href="#">FBlockID</a> , <a href="#">InstID</a>
		Error	ErrorCode, ErrorInfo

### 2.1.43.2 Parameter

#### SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### FBlockID

ID of the FBlock that owns the source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### InstID

Instance ID of the FBlock instance that owns the source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

#### SourceNr

Number of a data source (within one FBlock there can be more than one), for example, 0x01 for the first source.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## SinkNr

Number of a data sink (within one FBlock there can be more than one), for example, 0x01 for the first sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## PacketFormat

PacketFormat gives the packet format that is used by the source (please see MOST Content Protection Scheme--DTCP-Implementation).

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x04	0x00	NotDefined	Not defined
		0x01	NotApplicable	Not applicable
		0x02	GenericMOSTDTCP	Generic MOST-DTCP Format
		0x03	AVPacketizedMOSTDTC P	A/V Packetized MOST-DTCP Format
		0x04	AVPacketizedDTCPPI P	A/V Packetized DTCP-IP; deprecated!

## MediaType

This parameter refers to the MediaType values, which are given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## Packetlength

This parameter refers to the MOST Packet Length value, which is given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## EncryptionFrameSize

This parameter refers to the DTCP Encryption Frame Size value, which is given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## 2.1.44 DTCP\_InfoExt (0x126)

Occurrence: Conditional

The presence of this function is conditional; it is required for FBlocks that support Digital Transmission Content Protection. This function gives information about MOST DTCP parameters.

### 2.1.44.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_InfoExt (0x126)	Get	<a href="#">SourceSinkNr</a>
		Status	<a href="#">SourceSinkNr</a> , <a href="#">PacketFormat</a> , <a href="#">Packetlength</a> , <a href="#">EncryptionFrameSize</a> , <a href="#">EMI_Byte</a> , <a href="#">NumInfoBytesFollowing</a> , <a href="#">MediaType</a> , <a href="#">RemainingInfoBytes</a>
		Error	ErrorCode, ErrorInfo

### 2.1.44.2 Parameter

#### SourceSinkNr

Number of a data source or sink.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

#### PacketFormat

PacketFormat gives the packet format which is used by the source / sink (please see MOST Content Protection Scheme--DTCP Implementation).

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x04	0x00	NotDefined	Not defined
		0x01	NotApplicable	Not applicable
		0x02	GenericMOSTDTCP	Generic MOST-DTCP Format
		0x03	AVPacketizedMOSTDTC-P	A/V Packetized MOST-DTCP Format
		0x04	AVPacketizedDTCP-IP	A/V Packetized DTCP-IP; deprecated!

#### Packetlength

This parameter refers to the MOST Packet Length value, which is given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## EncryptionFrameSize

This parameter refers to the DTCP Encryption Frame Size value, which is given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## EMI\_Byte

This parameter refers to the EMI\_Byte (EMI and Odd/Even bit), which is given in the DTCP header.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte			1	none

## NumInfoBytesFollowing

This parameter refers to the Info[0]:"Number of Info bytes following" values, which are given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte			1	none

## MediaType

This parameter refers to the Info[1]:"MediaType" values, which are given in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## RemainingInfoBytes

This parameter refers to Info[2] up to Info[n] (variable length) of the DTCP Info bytes, which are given in the MOST Specification for Stream Transmission.

Basis data type	Length	Condition	Description
Stream		-	Content: <a href="#">RemainingInfoByte</a> (repeated)

## RemainingInfoByte

One Info byte, as provided in the MOST Specification for Stream Transmission.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte			1	none

## 2.1.45 ScreenFormat (0x130)

Occurrence: Optional

Image format of the video screen.

### 2.1.45.1 Format of Function

**Function classes:** Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	ScreenFormat (0x130)	Get	-
		Status	<a href="#">ScreenFormat</a>
		Error	ErrorCode, ErrorInfo

### 2.1.45.2 Parameter

#### ScreenFormat

Enumeration of available image formats.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x02	0x00	Unknown	Unknown
		0x01	Format_4_3	4:3
		0x02	Format_16_9	16:9



## 2.1.46 VideoFrequency (0x131)

Occurrence: Optional

The frame rate of the video signal. This function is available only if analog video output is available.

### 2.1.46.1 Format of Function

**Function classes:** Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	VideoFrequency (0x131)	Get	-
		Status	<a href="#">VideoFrequency</a>
		Error	ErrorCode, ErrorInfo

### 2.1.46.2 Parameter

#### VideoFrequency

Enumeration of available frequencies.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x02	0x00	unknown	unknown
		0x01	Freq50Hz	50 Hz
		0x02	Freq60Hz	60 Hz

## 2.1.47 VideoNorm (0x132)

Occurrence: Optional

The norm used for this video signal. This function is available only if analog video output is available.

### 2.1.47.1 Format of Function

**Function classes:** Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	VideoNorm (0x132)	Get	-
		Status	<a href="#">VideoNorm</a>
		Error	ErrorCode, ErrorInfo

### 2.1.47.2 Parameter

#### VideoNorm

Enumeration of the available television systems.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x2A	0x00	unknown	unknown
		0x01	EIA	EIA
		0x02	CCIR	CCIR
		0x09	NTSC	NTSC
		0x0A	NTSCEUROPE	NTSC-EUROPE
		0x0B	NTSCM	NTSC-M
		0x0C	NTSCJAPAN	NTSC-JAPAN
		0x13	PAL	PAL
		0x14	PALBG	PAL-BG
		0x15	PALI	PAL-I
		0x16	PALM	PAL-M
		0x17	PALN	PAL-N
		0x18	PALDK	PAL-DK
		0x19	PALAUSTRALIA	PAL-AUSTRALIA
		0x1A	PALITALIA	PAL-ITALIA
		0x1B	PALMAROCCO	PAL-MAROCCO
		0x1C	PALVRC	PAL-VRC
		0x1E	SECAMBG	SECAM-BG
		0x1F	SECAMDK	SECAM-DK
		0x20	SECAMK1	SECAM-K1
		0x21	SECAML	SECAM-L
		0x28	HDTV	HDTV
		0x29	MUSEJAPAN	MUSE(JAPAN)
		0x2A	HDTVUSA	HDTV(USA)

## 2.1.48 VideoSignalFormat (0x133)

Occurrence: Optional

Information about the format of the video signal

### 2.1.48.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	VideoSignalFormat (0x133)	Get	-
		Status	<a href="#">VideoSignalFormat</a>
		Error	ErrorCode, ErrorInfo

### 2.1.48.2 Parameter

#### VideoSignalFormat

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x0C	0x00	unknown	unknown
		0x01	BAS	BAS
		0x0A	FBASCVBS	FBAS (CVBS)
		0x0B	YC	Y/C
		0x0C	RGB	RGB

## 2.1.49 VideoFormat (0x135)

Occurrence: Optional

The format of the video signal. This function is available only if analog video output is available.

### 2.1.49.1 Format of Function

**Function classes:** Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	VideoFormat (0x135)	Set	<a href="#">VideoFormat</a>
		Get	-
		SetGet	<a href="#">VideoFormat</a>
		Status	<a href="#">VideoFormat</a>
		Error	ErrorCode, ErrorInfo

### 2.1.49.2 Parameter

#### VideoFormat

Enumeration of the available video appearance settings.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x08	0x00	unknown	unknown
		0x01	Standard4_3	Standard 4:3
		0x02	Standard16_9	Wide 16:9
		0x03	Letterbox	Letter box
		0x04	PanSCAN	Pan SCAN
		0x05	Cinemascope	Cinemascope
		0x06	reserved1	reserved
		0x07	reserved2	reserved
		0x08	reserved3	reserved

## 2.1.50 DeckStatus (0x200)

Occurrence: Optional

This property controls and shows the state of the drive. For connectivity devices, this property shows the state of the connected device; it assumes that a device is connected and therefore does not make any statement about the connectivity status. The default value is 'Load / Connect'.

### 2.1.50.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	DeckStatus (0x200)	Set	<a href="#">DeckStatus</a>
		Get	-
		SetGet	<a href="#">DeckStatus</a>
		Status	<a href="#">DeckStatus</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.50.2 Parameter

#### DeckStatus

- 0x00...0x1F = general states
- 0x20...0x2F = video specific states
- 0x30...0x3F = tape specific states
- 0x40...0x4F = file handling
- 0x50...0x5F = recording
- 0x60...0xFF = miscellaneous

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0xFF	0x00	Play	Play
		0x01	Stop	Stop
		0x02	Pause	Pause
		0x03	LoadConnect	Load / Connect
		0x04	UnloadDisconnect	Unload / Disconnect
		0x05	SearchForward	Search Forward
		0x06	SearchBackward	Search Backward
		0x07	FastForward	Fast Forward
		0x08	FastBackward	Fast Backward
		0x09	Empty	Empty / No titles available
		0x0A	Retract	Retract
		0x20	SlowMotionForward	Slow Motion Forward
		0x21	SlowMotionBackward	Slow Motion Backward
		0x22	StepbyStep	Step by Step
		0x23	PreStop	PreStop
		0x30	RewindBeginTape	Rewind to Begin of Tape
		0x31	ForwardEndTape	Forward to End of Tape

Basis data type	Range of values	Code	Symbolic Name	Description
		0x32	SearchStartPositionNext	Search Startposition next title
		0x33	SearchStartPositionLast	Search Startposition last Title
		0x40	Fileplay	Fileplay
		0x41	Filetransfer	Filetransfer
		0x50	Record	Record
		0x51	TimeshiftPlay	Timeshift Play
		0x52	TimeshiftPause	Timeshift Pause
		0x53	TimeshiftSearchForward	Timeshift Search Forward
		0x54	TimeshiftSearchBackward	Timeshift Search Backward
		0x55	TimeshiftFastForward	Timeshift Fast Forward
		0x56	TimeshiftFastBackward	Timeshift Fast Backward
		0x57	TimeshiftSlowMotionForward	Timeshift Slow Motion Forward
		0x58	TimeshiftSlowMotionBackward	Timeshift Slow Motion Backward
		0x59	TimeshiftStepbyStep	Timeshift StepbyStep
		0x60	SkipForward	Skip Forward
		0x61	SkipBackward	Skip Backward
		0x62	SlotReady	Slot ready for medium insertion
		0xFF	ERROR	ERROR

## ErrorCode

(Refer to the MOST Specification, section "Structure of MOST Messages: OPType")

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ErrorInfo

Besides the error information provided in the MOST Specification, additional function specific error info values are specified here.

Basis data type	Length	Condition	Description
Stream		ErrorCode = 0x20	Content: <a href="#">ErrorInfo_Detail</a>

## ErrorInfo\_Detail

Error info details.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x20...0x20	0x20	UOP	user operation prohibition

## 2.1.51 TimePosition (0x201)

Occurrence: Optional

Display of current or setting of a new position in milliseconds.

Datatype: Record of {DiskTime,TrackTime, TitleTime}

**Note:** An event is triggered only every second.

### 2.1.51.1 Format of Function

**Function classes:** Record of { Number Number Number }

FBlock	Function	OPType	Parameter
GeneralFBlock	TimePosition (0x201)	Set	<a href="#">Pos</a> , <a href="#">Data</a>
		Get	<a href="#">Pos</a>
		SetGet	<a href="#">Pos</a> , <a href="#">Data</a>
		Increment	<a href="#">Pos</a> , <a href="#">NSteps</a>
		Decrement	<a href="#">Pos</a> , <a href="#">NSteps</a>
		Status	<a href="#">Pos</a> , <a href="#">Data</a>
		Error	ErrorCode, ErrorInfo

### 2.1.51.2 Parameter

#### Pos

The parameter Pos={x,y} consists of two bytes, x and y, and shows which parameter shall be set, queried or read. Since this is a unidimensional construction, the second Byte y is unused (y=0=const) and the simplified notation Pos={x} is valid.

Valid range: x=0..3

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0	0...3	1	none

#### Data

The content of Data depends on parameter Pos={x,y}.

Basis data type	Length	Description	
Record	-	Pos	Data
		{ x==0 }	{DiskTime, TrackTime, TitleTime}
		{ x=1 }	{DiskTime}
		{ x=2 }	{TrackTime}
		{ x=3 }	{TitleTime}

#### DiskTime

Time details in milliseconds related to start of disk.

Basis data type	Exp.	Range of values	Step	Unit
Signed Long	0		1	ms

## TrackTime

---

Time details in milliseconds related to start of current track.

Basis data type	Exp.	Range of values	Step	Unit
Signed Long	0		1	ms

## TitleTime

---

Time details in milliseconds related to the start of the current title.

Basis data type	Exp.	Range of values	Step	Unit
Signed Long	0		1	ms

## NSteps

---

Number of steps for adjustment.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none



## 2.1.52 TrackPosition (0x202)

Occurrence: Optional

Used to query the current track or to play a different track.

**Note:**

- Track=0 for "no track", e.g., if there is no medium available.
- For a tape player, Track=1 corresponds to the first side of the tape and Track=2 corresponds to the second side.
- The currently valid range of values is dependent on the medium. When required, the actual range is queriable by the interface.

### 2.1.52.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	TrackPosition (0x202)	Set	<a href="#">Track</a>
		Get	-
		SetGet	<a href="#">Track</a>
		Increment	<a href="#">NSteps</a>
		Decrement	<a href="#">NSteps</a>
		Status	<a href="#">Track</a>
		Error	ErrorCode, ErrorInfo

### 2.1.52.2 Parameter

#### Track

The current track.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### NSteps

Number of steps for adjustment.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## 2.1.53 FramePosition (0x203)

Occurrence: Optional

Query the current playback position in frame units or continue playback from a different frame.

### 2.1.53.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	FramePosition (0x203)	Set	<a href="#">Frame</a>
		Get	-
		SetGet	<a href="#">Frame</a>
		Increment	<a href="#">NSteps</a>
		Decrement	<a href="#">NSteps</a>
		Status	<a href="#">Frame</a>
		Error	ErrorCode, ErrorInfo

### 2.1.53.2 Parameter

#### Frame

The current frame.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Long	0		1	none

#### NSteps

Number of steps for adjustment.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## 2.1.54 TitlePosition (0x205)

Occurrence: Optional

Used to query the current sequence or continue playback from a different sequence.

### 2.1.54.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	TitlePosition (0x205)	Set	<a href="#">TitlePosition</a>
		Get	-
		SetGet	<a href="#">TitlePosition</a>
		Increment	<a href="#">NSteps</a>
		Decrement	<a href="#">NSteps</a>
		Status	<a href="#">TitlePosition</a>
		Error	ErrorCode, ErrorInfo

### 2.1.54.2 Parameter

#### TitlePosition

Title position.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### NSteps

Number of steps for adjustment.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## 2.1.55 ChapterPosition (0x206)

Occurrence: Optional

Used to query the current chapter or continue playback from a different chapter.

### 2.1.55.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	ChapterPosition (0x206)	Set	<a href="#">ChapterPosition</a>
		Get	-
		SetGet	<a href="#">ChapterPosition</a>
		Increment	<a href="#">NSteps</a>
		Decrement	<a href="#">NSteps</a>
		Status	<a href="#">ChapterPosition</a>
		Error	ErrorCode, ErrorInfo

### 2.1.55.2 Parameter

#### ChapterPosition

Number of steps for adjustment.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### NSteps

Number of steps for adjustment.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...255	1	none

## 2.1.56 DeckStatusExt (0x207)

Occurrence: Optional

DeckStatusExt extends the DeckStatus function with an additional parameter, which may be used to identify the HMI that issued a command.

This property controls and shows the state of the drive. For connectivity devices, this property shows the state of the connected device; it assumes that a device is connected and therefore does not make any statement about the connectivity status. The default value is 'Load / Connect'.

### 2.1.56.1 Format of Function

**Function classes:** Sequence Property

FBlock	Function	OPType	Parameter
GeneralFBlock	DeckStatusExt (0x207)	Set	<a href="#">DeckStatus</a> , <a href="#">TransactionID</a>
		Get	-
		SetGet	<a href="#">DeckStatus</a> , <a href="#">TransactionID</a>
		Status	<a href="#">DeckStatus</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.56.2 Parameter

#### DeckStatus

- 0x00...0x1F = general states
- 0x20...0x2F = video specific states
- 0x30...0x3F = tape specific states
- 0x40...0x4F = file handling
- 0x50...0x5F = recording
- 0x60...0xFF = miscellaneous

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0xFF	0x00	Play	Play
		0x01	Stop	Stop
		0x02	Pause	Pause
		0x03	LoadConnect	Load / Connect
		0x04	UnloadDisconnect	Unload / Disconnect
		0x05	SearchForward	Search Forward
		0x06	SearchBackward	Search Backward
		0x07	FastForward	Fast Forward
		0x08	FastBackward	Fast Backward
		0x09	Empty	Empty / No titles available
		0x0A	Retract	Retract
		0x20	SlowMotionForward	Slow Motion Forward
		0x21	SlowMotionBackward	Slow Motion Backward
		0x22	StepbyStep	Step by Step

Basis data type	Range of values	Code	Symbolic Name	Description
		0x23	PreStop	PreStop
		0x30	RewindBeginTape	Rewind to Begin of Tape
		0x31	ForwardEndTape	Forward to End of Tape
		0x32	SearchStartPositionNext	Search Startposition next title
		0x33	SearchStartPositionLast	Search Startposition last Title
		0x40	Fileplay	Fileplay
		0x41	Filetransfer	Filetransfer
		0x50	Record	Record
		0x51	TimeshiftPlay	Timeshift Play
		0x52	TimeshiftPause	Timeshift Pause
		0x53	TimeshiftSearchForward	Timeshift Search Forward
		0x54	TimeshiftSearchBackward	Timeshift Search Backward
		0x55	TimeshiftFastForward	Timeshift Fast Forward
		0x56	TimeshiftFastBackward	Timeshift Fast Backward
		0x57	TimeshiftSlowMotionForward	Timeshift Slow Motion Forward
		0x58	TimeshiftSlowMotionBackward	Timeshift Slow Motion Backward
		0x59	TimeshiftStepbyStep	Timeshift StepbyStep
		0x60	SkipForward	Skip Forward
		0x61	SkipBackward	Skip Backward
		0x62	SlotReady	Slot ready for medium insertion
		0xFF	ERROR	ERROR

## TransactionID

The transaction ID is mainly used to identify the HMI terminal that issues a command. If the TransactionID is not used, it is set to 0x0. The transaction ID can be used, for example, as follows:

4 bit	ID of the calling instance (HMI/terminal)
8 bit	Incremental counter, used to number the commands. 8 bits are the minimum width.
4 bit	Reserved (may be used for enhancing the counter)

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0	0x0...0xFFFF	1	none

## ErrorCode

(Refer to the MOST Specification, section "Structure of MOST Messages: OPType")

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ErrorInfo

---

Besides the error information provided in the MOST Specification, additional function specific error info values are specified here.

Basis data type	Length	Condition	Description
Stream		ErrorCode = 0x20	Content: <a href="#">ErrorInfo_Detail</a>

## ErrorInfo\_Detail

---

Error info details.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x20...0x20	0x20	UOP	user operation prohibition

## 2.1.57 VideoInteraction (0x251)

Occurrence: Optional

Operates the DVD menu.

### 2.1.57.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	VideoInteraction (0x251)	Start	<a href="#">VideoInteraction</a> , <a href="#">TouchPosX</a> , <a href="#">TouchPosY</a>
		StartResult	<a href="#">VideoInteraction</a> , <a href="#">TouchPosX</a> , <a href="#">TouchPosY</a>
		StartResultAck	<a href="#">SenderHandle</a> , <a href="#">VideoInteraction</a> , <a href="#">TouchPosY</a> , <a href="#">TouchPosX</a>
		StartAck	<a href="#">SenderHandle</a> , <a href="#">VideoInteraction</a> , <a href="#">TouchPosX</a> , <a href="#">TouchPosY</a>
		ErrorAck	<a href="#">SenderHandle</a> , <a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>
		ProcessingAck	<a href="#">SenderHandle</a>
		Processing	-
		Result	-
		ResultAck	<a href="#">SenderHandle</a>
		Error	<a href="#">ErrorCode</a> , <a href="#">ErrorInfo</a>

### 2.1.57.2 Parameter

#### VideoInteraction

Enumeration of the available commands for user interaction.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x12	0x00	Title	Title
		0x01	Root	Root
		0x02	Enter	Enter
		0x03	Up	Up
		0x04	Down	Down
		0x05	Right	Right
		0x06	Left	Left
		0x07	Back	Back
		0x08	PreviousChapter	Previous Chapter
		0x09	NextChapter	Next Chapter
		0x0A	Subtitle	Subtitle
		0x0B	Audio	Audio
		0x0C	Angle	Angle
		0x0D	PTT	PTT
		0x0E	Resume	Resume
		0x0F	KaraokeAudio	KaraokeAudio
		0x10	TouchEventPRESS	TouchEvent PRESS (use TouchPosX and TouchPosY parameters!)
		0x11	TouchEventRELEASE	TouchEvent RELEASE (use TouchPosX and TouchPosY)



Basis data type	Range of values	Code	Symbolic Name	Description
		0x12	TouchEventHOVER	TouchEvent HOVER (use TouchPosX and TouchPosY parameters!)

## TouchPosX

Touchscreen event X coordinate (only used if the VideoInteraction parameter is set to "TouchEvent DOWN", "TouchEvent UP", or "TouchEvent HOVER". Otherwise, set TouchPosX=0)

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	pixel

## TouchPosY

Touchscreen event Y coordinate (only used if the VideoInteraction parameter is set to "TouchEvent DOWN", "TouchEvent UP", or "TouchEvent HOVER". Otherwise, set TouchPosY=0)

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	pixel

## SenderHandle

Unique identifier of the requesting task within the device.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

## ErrorCode

(Refer to the MOST Specification, section "Structure of MOST Messages: OPType")

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## ErrorInfo

Besides the error information provided in the MOST Specification, additional function specific error info values are specified here.

Basis data type	Length	Condition	Description
Stream		ErrorCode = 0x20	Content: <a href="#">ErrorInfo_Detail</a>

## ErrorInfo\_Detail

Error info details.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x20...0x20	0x20	UOP	user operation prohibition

## 2.1.58 PlayerRegion (0x270)

Occurrence: Optional

This function should be used to read the current setting of the region code for the current player.

### 2.1.58.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	PlayerRegion (0x270)	Set	<a href="#">RegionCode</a>
		SetGet	<a href="#">RegionCode</a>
		Status	<a href="#">RegionCode</a> , <a href="#">NumberOfRemainingAttempts</a>
		Error	ErrorCode, ErrorInfo

### 2.1.58.2 Parameter

#### RegionCode

This parameter specifies the region code (Regional Playback Control) for the DVD.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0xFF	0x00	RegionCode0	"playing in all regions" or RegionCode not readable (e.g., not a Video DVD) or RC of DVD matches RC of system (for players not able to deliver real RC)
		0x01	RegionCode1	Bermuda, Canada, United States and U.S. territories
		0x02	RegionCode2	The Middle East, Western Europe, Central Europe, Egypt, French overseas territories, Greenland, Japan, Lesotho, South Africa and Swaziland
		0x03	RegionCode3	Southeast Asia, Hong Kong, Macau, south Korea and Taiwan
		0x04	RegionCode4	Australia, New Zealand, Central America, the Caribbean, Mexico, Oceania and South America (sans French Guyana)
		0x05	RegionCode5	The rest of Africa, Former Soviet Union, the Indian subcontinent, Mongolia and North Korea
		0x06	RegionCode6	Mainland China
		0x07	RegionCode7	Reserved for future use (found in use on

Basis data type	Range of values	Code	Symbolic Name	Description
				protected screener copies of MPAA-related DVDs, and "media-copies" of pre-releases in Asia)
		0x08	RegionCode8	International venues such as aircraft, cruise ships, etc.
		0xFF	Mismatch	RC of DVD-V does not match RC of System (for players not able to deliver real RC)

### NumberOfRemainingAttempts

This parameter specifies the number of remaining attempts to set the region code property.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x06	0x00	Permanent	Permanent
		0x01	LastChange	Last Change
		0x02	Set3Times	Set 3 times
		0x03	Set2Times	Set 2 times
		0x04	Set1Time	Set 1 time
		0x05	NeverSet	Never set
		0x06	SystemwideCounter	Systemwide counter (Player does not have its own counter)

## 2.1.59 DeckEvent (0x430)

Occurrence: Optional

Events concerning the drive or device. Whether the connected device/drive can signal the events to the gateway depends on the characteristics of the external device.

- OverTemperature / UnderTemperature => external device signals temperature problems
- OverVoltage / UnderVoltage => external device signals power problems
- TooMuchPowerDrawn => external device draws too much power

### 2.1.59.1 Format of Function

Function classes: Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	DeckEvent (0x430)	Get	-
		Status	<a href="#">DeckEvent</a>
		Error	ErrorCode, ErrorInfo

### 2.1.59.2 Parameter

#### DeckEvent

Type of event.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x06	0x00	NormalOperation	Normal Operation
		0x01	DeckError	Deck Error
		0x02	OverNormalTemperature	Over normal Temperature
		0x03	UnderNormalTemperature	Under normal Temperature
		0x04	UnderVoltage	Under Voltage
		0x05	OverVoltage	Over Voltage
		0x06	TooMuchPowerDrawn	Too much power drawn

## 2.1.60 MediaEvent (0x431)

Occurrence: Optional

Events concerning the current medium that occur during playback of tracks. The usage and generation of events is deployment specific. As rule of thumb the following recommendation is given: The implementation should avoid very fast toggling between different values and should therefore implement a minimum time of status hold time (e.g., 200 ms).

- NormalOperation: set as default if none of the other events are present
- Corrupted File or Track: signaled if the current track/title could not be played due to corruption (e.g., bit errors or unknown DRM format); an implementation specific recovery strategy may apply (e.g., jump to next track/title or stop playback)
- File or Track Change Delayed: should be signaled if track change is delayed (time is implementation specific)
- EndOfTracklist: Signaled if tracklist has reached the end and DeckStatus changes to "Pause" or "Stop"
- SecuredFile: signaled if file is secured (e.g., DRM protection recognized) and file could not be played
- File/Track not found: signaled if a track inside a playlist (e.g., m3u file) is not found; the recovery strategy is supplier-specific.
- Filter with no Media File/Track: signaled if the set filter (SelectMediaListFilter) covers no media files (e.g., empty directory or directory with no media files)

### 2.1.60.1 Format of Function

Function classes: Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	MediaEvent (0x431)	Get	-
		Status	<a href="#">MediaEvent</a>
		Error	ErrorCode, ErrorInfo

### 2.1.60.2 Parameter

#### MediaEvent

Type of event

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x12	0x00	NormalOperation	Normal Operation
		0x01	CorruptedFileOrTrack	Corrupted File or Track
		0x02	EndOfDisc	End of disc
		0x03	CorruptedROMFilesystem	Corrupted ROM Filesystem
		0x04	Startposition	Startposition
		0x05	TOCUnreadable	TOC Unreadable
		0x06	DiscNotAvailable	Disc not available
		0x07	EndOfFile	End of File
		0x08	FileOrTrackChangeDelay	File or Track Change

Basis data type	Range of values	Code	Symbolic Name	Description
			ed	Delayed
		0x09	EndOfTracklist	End of Tracklist
		0x0A	EndOfRandom	End of Random
		0x0B	NotReady	Not ready
		0x0C	TooManyConsecutiveErrors	Too many consecutive errors
		0x0D	UnsupportedFile	Unsupported file
		0x0E	Reserved1	reserved
		0x0F	Reserved2	reserved
		0x10	SecuredFile	Secured file (e.g., DRM)
		0x11	FileTrackNotFound	File/Track not found
		0x12	FilterWithNoMediaFile	Filter with no MediaFile/Track

## 2.1.61 Random (0x450)

Occurrence: Optional

This property is for switching on and off the Random function.

**Note:**

Depending on the value of RandomState, different scopes are covered.

- Disk: Random selection within the current medium
- Magazine: Random selection within the current magazine
- All Magazines: Random selection within all magazines

### 2.1.61.1 Format of Function

**Function classes:** Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	Random (0x450)	Set	<a href="#">RandomState</a>
		Get	-
		SetGet	<a href="#">RandomState</a>
		Status	<a href="#">RandomState</a>
		Error	ErrorCode, ErrorInfo

### 2.1.61.2 Parameter

#### RandomState

Depending on the player only a subset is available:

- AudioDiskPlayer (SinglePlayer): 0x00, 0x02
- AudioDiskPlayer(Changer): 0x00, 0x02, 0x03
- MultiMediaPlayer: 0x00, 0x02, 0x03

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x04	0x00	Off	Off
		0x01	Reserved	reserved
		0x02	Disk	Disk
		0x03	Magazine	Magazine (only drive with magazine)
		0x04	AllMagazines	All Magazines (only drive with several magazines)

## 2.1.62 Scan (0x451)

Occurrence: Optional

This property is for switching on and off the Scan function.

**Note:**

Depending on the value of ScanState, different scopes are covered.

- Disk: Seek within the current medium
- Magazine: Seek within the current magazine
- All Magazines: Seek within all magazines

### 2.1.62.1 Format of Function

**Function classes:** Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	Scan (0x451)	Set	<a href="#">ScanState</a>
		Get	-
		SetGet	<a href="#">ScanState</a>
		Status	<a href="#">ScanState</a>
		Error	ErrorCode, ErrorInfo

### 2.1.62.2 Parameter

#### ScanState

Depending on the player only a subset is available:

- AudioDiskPlayer (SinglePlayer): 0x00, 0x02
- AudioDiskPlayer (Changer): 0x00, 0x02, 0x03
- MultiMediaPlayer: 0x00, 0x02, 0x03

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x04	0x00	Off	Off
		0x01	Reserved	reserved
		0x02	Disk	Disk
		0x03	Magazine	Magazine (only drive with magazine)
		0x04	AllMagazines	All Magazines (only drive with several magazines)



## 2.1.63 Repeat (0x452)

Occurrence: Optional

This property is for switching on and off the repeat function. Note:

- Track: automatic repeat of the current track
- Disc: automatic repeat of the current medium
- Magazine: automatic repeat of all media of the current magazine
- All / All Magazines: automatic repeat of all tracks on the medium or magazines. Has a side effect that the Filter is set to contain all possible media titles.
- Chapter: automatic repeat of the current chapter
- Title: automatic repeat of the current title. Note that this does not influence the active Filter.
- AB: automatic repeat of the marked section.
- Folder: automatic repeat within the current folder/directory (optional). Requires a folder as the active filter. Reduces the active Filter to contain this directory only (without subdirectories!)
- SubFolder: automatic repeat within the current folder/directory including subfolder (optional). Requires a folder as the active filter.
- Tracklist: automatic repeat of the currents selected tracklist

It is advised to switch between Repeat Tracklist and Repeat Off only. Other repeat states are kept for backwards compatibility but are difficult to handle, since they either cause side effects or it is unclear what their scope is.

### 2.1.63.1 Format of Function

Function classes: Enumeration

FBlock	Function	OPType	Parameter
GeneralFBlock	Repeat (0x452)	Set	<a href="#">RepeatState</a>
		Get	-
		SetGet	<a href="#">RepeatState</a>
		Status	<a href="#">RepeatState</a>
		Error	ErrorCode, ErrorInfo

### 2.1.63.2 Parameter

#### RepeatState

Setting the RepeatState to "All" can cause a implicit change of the used filter for playback, if the filter was not set to all media on the connected device. The changed filter can be signaled by the gateway using notification for CurrentMediaListFilter property.

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x0C	0x00	Off	Off
		0x01	Track	Track
		0x02	Disk	Disk
		0x03	Magazine	Magazine

Basis data type	Range of values	Code	Symbolic Name	Description
		0x04	All	All / All magazines
		0x05	Chapter	Chapter
		0x06	Title	Title
		0x07	AB	AB
		0x0A	Folder	Folder
		0x0B	SubFolder	SubFolder
		0x0C	Tracklist	Tracklist

## 2.1.64 NextTrackToPlay (0x453)

Occurrence: Optional

Displays the current or sets a new position as track, which is to be played after the current track.  
Datatype: Record of {Track, MagazinePos, MagazineNo}

### 2.1.64.1 Format of Function

**Function classes:** Record of { Number Number Number }

FBlock	Function	OPType	Parameter
GeneralFBlock	NextTrackToPlay (0x453)	Set	<a href="#">Pos</a> , <a href="#">Data</a>
		Get	<a href="#">Pos</a>
		SetGet	<a href="#">Pos</a> , <a href="#">Data</a>
		Status	<a href="#">Pos</a> , <a href="#">Data</a>
		Error	ErrorCode, ErrorInfo

### 2.1.64.2 Parameter

#### Pos

The parameter Pos={x,y} consists of two bytes, x and y, and shows which parameter shall be set or read. Since this is a unidimensional construction, the second byte y is unused (y=0=const) and the simplified notation Pos={x} is valid.  
Valid range: x=0..3

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### Data

The content of Data depends on parameter Pos={x,y}.

Basis data type	Length	Description	
Record	-	Pos	Data
		{ x==0 }	{Track, MagazinePos, MagazineNo}

#### Track

Track number.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	none

#### MagazinePos

Position within the magazine.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## MagazineNo

---

Number of the magazine.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

## 2.1.65 Deemphasis (0x454)

Occurrence: Optional

Switches the deemphasis filter for disc drives on or off.

### 2.1.65.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
GeneralFBlock	Deemphasis (0x454)	Get	-
		Status	<a href="#">OnOff</a>
		Error	ErrorCode, ErrorInfo

### 2.1.65.2 Parameter

#### OnOff

Basis data type	Bit #	Code	Description
Boolean	Bit 0	True	On
		False	Off

## 2.1.66 SlowFwSpeed (0x455)

Occurrence: Optional

Displays the current speed or sets the new speed for "slow motion forward".

### 2.1.66.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	SlowFwSpeed (0x455)	Set	<a href="#">SlowSpeed</a>
		Get	-
		SetGet	<a href="#">SlowSpeed</a>
		Status	<a href="#">SlowSpeed</a>
		Error	ErrorCode, ErrorInfo

### 2.1.66.2 Parameter

#### SlowSpeed

Speed indicator as inverse value (e.g., SlowSpeed=4: Slow motion with 1/4 of normal speed).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	2...16	1	none

## 2.1.67 SlowBwSpeed (0x456)

Occurrence: Optional

Displays the current speed or sets the new speed for "slow motion backward".

### 2.1.67.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	SlowBwSpeed (0x456)	Set	<a href="#">SlowSpeed</a>
		Get	-
		SetGet	<a href="#">SlowSpeed</a>
		Status	<a href="#">SlowSpeed</a>
		Error	ErrorCode, ErrorInfo

### 2.1.67.2 Parameter

#### SlowSpeed

Speed indicator as inverse value (e.g., SlowSpeed=4: Slow motion with 1/4 of normal speed).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	2...16	1	none

## 2.1.68 FastFwSpeed (0x457)

Occurrence: Optional

Displays the current speed or sets the new speed for "fast motion forward".

### 2.1.68.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	FastFwSpeed (0x457)	Set	<a href="#">FastSpeed</a>
		Get	-
		SetGet	<a href="#">FastSpeed</a>
		Status	<a href="#">FastSpeed</a>
		Error	ErrorCode, ErrorInfo

### 2.1.68.2 Parameter

#### FastSpeed

Speed value (e.g., FastSpeed=4: Fast motion with 4 \* normal speed).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	2...32	1	none



## 2.1.69 FastBwSpeed (0x458)

Occurrence: Optional

Displays the current speed or sets the new speed for "fast motion backward".

### 2.1.69.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
GeneralFBlock	FastBwSpeed (0x458)	Set	<a href="#">FastSpeed</a>
		Get	-
		SetGet	<a href="#">FastSpeed</a>
		Status	<a href="#">FastSpeed</a>
		Error	ErrorCode, ErrorInfo

### 2.1.69.2 Parameter

#### FastSpeed

Speed value (e.g., FastSpeed=4: Fast motion with 4 \* normal speed).

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	2...32	1	none