

MOST

Media Oriented Systems Transport

Multimedia and Control
Networking Technology

**MOST FBlock Template GeneralFBlock –
Speed Grade MOST25
Rev 2.5.1
10/2007**

MOSTCO CONFIDENTIAL

See page 3 for the terms of disclosure



Legal Notice

COPYRIGHT

© Copyright 1999 - 2007 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
Bannwaldallee 48
D-76185 Karlsruhe
Germany

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

Fax: (+49) (0) 721 966 50 01

E-mail: contact@mostcooperation.com

Web: 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 - 2007 MOST Cooperation
All rights reserved

MOST is a registered trademark

1	INTRODUCTION	8
2	FBLOCK DEFINITION	8
2.1	GeneralFBlock	8
2.1.1	FktIDs (0x000)	10
2.1.2	Notification (0x001)	10
2.1.3	NotificationCheck (0x002)	12
2.1.4	AsyncControlSwitch (0x003)	13
2.1.5	Version (0x010)	14
2.1.6	DynArrayIns (0x080)	15
2.1.7	DynArrayDel (0x081)	16
2.1.8	MapIns (0x082)	17
2.1.9	MapDel (0x083)	19
2.1.10	CreateArrayWindow (0x090)	20
2.1.11	DestroyArrayWindow (0x091)	21
2.1.12	MoveArrayWindow (0x092)	22
2.1.13	SearchArrayWindow (0x093)	23
2.1.14	LongArrayInfo (0x094)	24
2.1.15	ArrayWindowIns (0x09A)	25
2.1.16	ArrayWindowDel (0x09B)	27
2.1.17	PowerDownDelay (0x0A0)	28
2.1.18	SourceInfo (0x100)	29
2.1.19	Allocate (0x101)	31
2.1.20	DeAllocate (0x102)	32
2.1.21	SourceActivity (0x103)	32
2.1.22	SourceName (0x104)	33
2.1.23	SourceConnect (0x105)	33
2.1.24	SourceDisConnect (0x106)	35
2.1.25	SourceRouting (0x107)	35
2.1.26	SinkInfo (0x110)	36
2.1.27	Connect (0x111)	38
2.1.28	DisConnect (0x112)	39
2.1.29	Mute (0x113)	40
2.1.30	SinkName (0x114)	41
2.1.31	ConnectTo (0x115)	41
2.1.32	SyncDataInfo (0x116)	42
2.1.33	SinkRouting (0x117)	43
2.1.34	DTCP_StartProcess (0x120)	44
2.1.35	DTCP_Control (0x121)	45
2.1.36	DTCP_Status (0x122)	46
2.1.37	DTCP_CipherStatus (0x123)	47
2.1.38	DTCP_Info (0x124)	48
2.1.39	DTCP_ContentKeyProcess (0x125)	49

References

Number	Document
[1]	MOST Specification 2V5

Bibliography MOST Library

This is a list of released FBlocks and FBlock templates at the release time of this specification. FBlocks which are released later are not reflected in this list.

FBlockID	FunctionBlock
-	GeneralFBlock
-	GeneralPlayer
0x01	NetBlock
0x02	NetworkMaster
0x03	ConnectionMaster
0x06	Diagnosis
0x0E	Tool
0x0F	Enhanced Testability
0x22	AudioAmplifier
0x24	AuxIn
0x26	MicrophoneInput
0x30	AudioTapePlayer
0x31	AudioDiskPlayer
0x34	DVDVideoPlayer
0x40	AmFmTuner
0x41	TMCTuner
0x42	TVTuner
0x43	DABTuner
0x44	SDARS
0x50	Telephone
0x51	GeneralPhoneBook
0x60	GraphicDisplay
-	Unique Functions

GeneralFBlock FBlock Speed Grade MOST25 Change History

Changes GeneralFBlock FBlock 2.5.0 to GeneralFBlock FBlock 2.5.1 - Speed Grade MOST25

Change Ref.	FktID	Changes
2.5.1-001	General	- Updated definitions section.
2.5.1-002	General	- Replaced unit references to "unit_not_defined" and "InvalidUnit" with "unit_none".
2.5.1-003	General	- Added range values for parameter BlockWidth (1-60).
2.5.1-004	General	- Corrected date format to comply with ISO 8601 standard (YYYY-MM-DD).
2.5.1-005	0x0A0	- Added function PowerDownDelay (0x0A0).
2.5.1-006	0x120	- Added function DTCP_StartProcess.
2.5.1-007	0x121	- Added function DTCP_Control.
2.5.1-008	0x122	- Added function DTCP_Status.
2.5.1-009	0x123	- Added function DTCP_CipherStatus.
2.5.1-010	0x124	- Added function DTCP_Info.
2.5.1-011	0x125	- Added function DTCP_ContentKeyProcess.

Changes GeneralFBlock FBlock 2.3.3 to GeneralFBlock FBlock 2.5.0

Change Ref.	FktID	Changes
2.5.0-001	General	- Removed FBlockID because GeneralFBlock is merely a template. - Many minor corrections in spelling and use of MOST terminology. - Amended XML storage format with new Function Classes, data types and units.
2.5.0-002	0x003	- Added function AsyncControlSwitch.
2.5.0-003	0x010	- Added function Version.
2.5.0-004	0x082	- Added function MapIns.
2.5.0-005	0x083	- Added function MapDel.
2.5.0-006	0x092	- Added ResultAck OPType and description for MoveArrayWindow (0x092)
2.5.0-007	0x093	- Added reference to MOST specification in description of SearchArrayWindow.
2.5.0-008	0x094	- Parameter FktIDAW of function LongArrayInfo changed to Unsigned Word.
2.5.0-009	0x100, 0x104, 0x110, 0x114	- Added note about multiple notifications to SourceInfo, SourceName, SinkInfo and SinkName
2.5.0-010	0x110	- Removed parameters ChannelList and SinkDelay from function SinkInfo because they were not used by any OPType. They are now embedded in the DataDescription stream. - Added missing ChannelList stream to DataDescription stream case 0x01.

Changes GeneralFBlock FBlock 2.3.2 to GeneralFBlock FBlock 2.3.3

Change Ref.	FktID	Changes
2.3.3-001	0x093	- Parameter FktIDMotherArray changed to FktIDArrayWindow

Changes GeneralFBlock FBlock 2.3.1 to GeneralFBlock FBlock 2.3.2

Change Ref.	FktID	Changes
2.3.2-001	0x002	- Changed description of parameter FktIDList.
		-
		-

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 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.

Function Overview		
FktID	Name	Section Type
0x000	FktIDs	Coordination
0x001	Notification	Coordination
0x002	NotificationCheck	Coordination
0x003	AsyncControlSwitch	Coordination
0x010	Version	Coordination
0x080	DynArrayIns	Coordination
0x081	DynArrayDel	Coordination
0x082	MapIns	Coordination
0x083	MapDel	Coordination
0x090	CreateArrayWindow	Coordination
0x091	DestroyArrayWindow	Coordination
0x092	MoveArrayWindow	Coordination
0x093	SearchArrayWindow	Coordination
0x094	LongArrayInfo	Coordination
0x09A	ArrayWindowIns	Coordination
0x09B	ArrayWindowDel	Coordination
0x0A0	PowerDownDelay	Coordination
0x100	SourceInfo	Coordination
0x101	Allocate	Coordination
0x102	DeAllocate	Coordination
0x103	SourceActivity	Coordination
0x104	SourceName	Coordination
0x105	SourceConnect	Coordination
0x106	SourceDisConnect	Coordination
0x107	SourceRouting	Coordination
0x110	SinkInfo	Coordination
0x111	Connect	Coordination
0x112	DisConnect	Coordination
0x113	Mute	Coordination
0x114	SinkName	Coordination
0x115	ConnectTo	Coordination
0x116	SyncDataInfo	Coordination
0x117	SinkRouting	Coordination
0x120	DTCP_StartProcess	Coordination
0x121	DTCP_Control	Coordination
0x122	DTCP_Status	Coordination
0x123	DTCP_CipherStatus	Coordination
0x124	DTCP_Info	Coordination
0x125	DTCP_ContentKeyProcess	Coordination

2.1.1 FktIDs (0x000)

Section type: Coordination

With the property FktIDs, the functions of an FBlock may be queried.

2.1.1.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	FktIDs (0x000)	Get	-
		Status	BitField
		Error	ErrorCode, ErrorInfo

2.1.1.2 Parameter

BitField

RLE-coded bitfield of available functions.

Remark: FktIDs are 12-bit encoded!

Basis datatype	Length	Condition	Description
Stream		-	FktID1, FktID2, ...

2.1.2 Notification (0x001)

Section type: Coordination

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	Control, DeviceID, FktIDList
		Get	FktID
		Status	FktID, DeviceIDList
		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 datatype	Range of values	Code	Description
Enum	0x00..0x03	0x00	SetAll
		0x01	SetFunction
		0x02	ClearAll
		0x03	ClearFunction

DeviceID

Either RxTxLog of a device or a group address.

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

FktIDList

List of functions. The maximum list length is 4.

Basis datatype	Length	Condition	Description
Stream		-	Content: FktID FktID {, FktID}

FktID

Function ID.

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

DeviceIDList

List of devices.

Basis datatype	Length	Condition	Description
Stream		-	Content: DeviceID DeviceID {, DeviceID}

2.1.3 NotificationCheck (0x002)

Section type: Coordination

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	DeviceID
		Status	DeviceID, FktIDList
		Error	ErrorCode, ErrorInfo

2.1.3.2 Parameter

DeviceID

Either RxTxLog of a device or a group address.

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

FktIDList

List of functions.

Basis datatype	Length	Condition	Description
Stream		-	Content: FktID FktID {, FktID}

FktID

Function ID.

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

2.1.4 AsyncControlSwitch (0x003)

Section type: Coordination

Due to the high amount of data that can be transferred when calling certain functions, this switch determines whether some or all OPTypes of certain functions use the Packet Data Channel instead of the Control Channel. See the function definitions to determine which functions and OPTypes are affected by this switch.

The hardware need not support the Packet Data Channel; in this case, AsyncControlSwitch will respond with an error. The default on every startup, Configuration.Status(NotOK), etc. must be "False". Before using the Packet Data Channel, the Controller must Set or SetGet this property to "True" and must receive a Status of "True" for confirmation. The Controller may Set or SetGet this property to "True" only if it supports the Packet Data Channel itself. If several Controllers access the same FBlock, it is advised to set a notification for this property and have each Controller send AsyncControlSwitch. Set either with "True" or "False" depending on the abilities of the Controller. This will ensure that switching to the Packet Data Channel is only done when all Controllers (and the FBlock) support packet data handling.

Notification must be possible.

2.1.4.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
GeneralFBlock	AsyncControl Switch (0x003)	Set	Async
		Get	-
		SetGet	Async
		Status	Async
		Error	ErrorCode, ErrorInfo

2.1.4.2 Parameter

Async

Determines whether requests and responses (status and error messages) for AudioListInfo and CurrentAudioListInfo are sent through the Packet Data Channel instead of the Control Channel.

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	True	Packet Data Channel
		False	Control Channel

2.1.5 Version (0x010)

Section type: Coordination

This function has to be implemented in every node. 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)

2.1.5.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	Version (0x010)	Get	-
		Status	Major, Minor, Build
		Error	ErrorCode, ErrorInfo

2.1.5.2 Parameter

Major

Major version value of the FBlock.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

Minor

Minor version value of the FBlock.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

Build

Build number of the FBlock.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

2.1.6 DynArrayIns (0x080)

Section type: Coordination

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: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DynArrayIns (0x080)	StartResult	SenderHandle, FktID, Tag, Quantity, InsertData
		Ack	
		AbortAck	SenderHandle, FktID, Tag, Quantity, InsertData
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		Processing Ack	SenderHandle
		ResultAck	SenderHandle

2.1.6.2 Parameter

SenderHandle

Unique identifier of a task.

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

FktID

Function ID of the DynamicArray.

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

Quantity

Number of rows.

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

InsertData

Data to be inserted.

Basis datatype	Length	Condition	Description
Stream		-	Content: Entry Entry {, Entry }

Entry

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

Basis datatype	Length	Description
Stream	-	

2.1.7 DynArrayDel (0x081)

Section type: Coordination

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: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DynArrayDel (0x081)	StartResult Ack	SenderHandle, FktID, Tag, Quantity
		AbortAck	SenderHandle, FktID, Tag, Quantity
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		Processing Ack	SenderHandle
		ResultAck	SenderHandle

2.1.7.2 Parameter

SenderHandle

Unique identifier of a task.

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

FktID

Function ID of the DynamicArray.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..4095	1	none

Tag

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

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

Quantity

Number of rows.

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

2.1.8 MapIns (0x082)

Section type: Coordination

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: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	MapIns (0x082)	StartResult	SenderHandle, FktID, Quantity, InsertData
		Ack	
		AbortAck	SenderHandle
		StartAck	SenderHandle, FktID, Quantity, InsertData
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		Processing Ack	SenderHandle
		ResultAck	SenderHandle, Quantity, TagList

2.1.8.2 Parameter

SenderHandle

Unique identifier of the requesting task within the device.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

FktID

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

Quantity

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	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 datatype	Length	Condition	Description
Stream		-	Content: Entry Entry {, Entry }

Entry

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

Basis datatype	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 datatype	Length	Condition	Description
Stream		-	Content: Tag Tag {, Tag }

Tag

A tag that identifies an entry of the datatype Map.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

2.1.9 MapDel (0x083)

Section type: Coordination

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: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	MapDel (0x083)	StartResultAck	SenderHandle, FktID, Quantity, TagList
		AbortAck	SenderHandle
		StartAck	SenderHandle, FktID, Quantity, TagList
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		ProcessingAck	SenderHandle
		ResultAck	SenderHandle, FktID, Quantity, TagList

2.1.9.2 Parameter

SenderHandle

Unique identifier of the requesting task within the device.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

FktID

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	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 datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

TagList

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

Basis datatype	Length	Condition	Description
Stream		-	Content: Tag Tag {, Tag }

Tag

A tag that identifies an entry of the datatype Map.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

2.1.10 CreateArrayWindow (0x090)

Section type: Coordination

Create a new instance of an ArrayWindow.

2.1.10.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	CreateArray Window (0x090)	StartResult Ack	SenderHandle, FktIDMotherArray, Tag, WindowSize
		AbortAck	SenderHandle, FktIDMotherArray, Tag, WindowSize
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		ResultAck	SenderHandle, FktIDArrayWindow

2.1.10.2 Parameter

SenderHandle

Unique identifier of a task.

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

FktIDMotherArray

FktID of the MotherArray.

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

WindowSize

Size of the ArrayWindow.

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

FktIDArrayWindow

FktID of the created ArrayWindow.

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

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

2.1.11 DestroyArrayWindow (0x091)

Section type: Coordination

Delete an ArrayWindow.

2.1.11.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DestroyArray Window (0x091)	StartResult Ack	SenderHandle, FktIDArrayWindow
		AbortAck	SenderHandle, FktIDArrayWindow
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		ResultAck	SenderHandle

2.1.11.2 Parameter

SenderHandle

Unique identifier of a task.

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

FktIDArrayWindow

FktID of the created ArrayWindow.

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

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

2.1.12 MoveArrayWindow (0x092)

Section type: Coordination

This function is used to position an ArrayWindow on a MotherArray.

2.1.12.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	MoveArrayWindow (0x092)	AbortAck	SenderHandle, FktIDArrayWindow, MovingMode, Number, Tag
		StartAck	SenderHandle, FktIDArrayWindow, MovingMode, Number, Tag
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo

2.1.12.2 Parameter

SenderHandle

Unique identifier of a task.

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

FktIDArrayWindow

Function ID of the ArrayWindow.

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

MovingMode

Determines the desired positioning mode for the ArrayWindow.

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	Top
		0x01	Bottom
		0x02	Up
		0x03	Down
		0x04	Absolute

Number

Number of lines to move the ArrayWindow.

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

2.1.13 SearchArrayWindow (0x093)

Section type: Coordination

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: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	SearchArray Window (0x093)	StartResult Ack	SenderHandle, FktIDArrayWindow, PosY, SearchString
		AbortAck	SenderHandle, FktIDArrayWindow, PosY, SearchString
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		Processing Ack	SenderHandle
		ResultAck	SenderHandle

2.1.13.2 Parameter

SenderHandle

Unique identifier of a task.

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

FktIDArrayWindow

Function ID of the ArrayWindow.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

PosY

PosY indicates which column is the target of the search.

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

SearchString

The text to search for.

Basis datatype	MaxSize
String	100

2.1.14 LongArrayInfo (0x094)

Section type: Coordination

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	Pos
		Status	Pos, Data
		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, queried or read.

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

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

FktIDAW

Function ID of an ArrayWindow.

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

DeviceIDController

DeviceID of the Controller that created the ArrayWindow.

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

2.1.15 ArrayWindowIns (0x09A)

Section type: Coordination

Inserts a number of lines into a MotherArray.

2.1.15.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	ArrayWindowIns (0x09A)	StartResult Ack	SenderHandle, FktIDArrayWindow, Tag, Quantity, InsertData
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		Processing Ack	SenderHandle
		ResultAck	SenderHandle

2.1.15.2 Parameter

SenderHandle

Unique identifier of a task.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

FktIDArrayWindow

FktID of the ArrayWindow.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	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 datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

Quantity

Number of rows.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

InsertData

Data to be inserted.

Basis datatype	Length	Condition	Description
Stream		-	Content: Entry Entry {, Entry }

Entry

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

Basis datatype	Length	Description
Stream	-	

2.1.16 ArrayWindowDel (0x09B)

Section type: Coordination

Deletes a number of lines from a MotherArray.

2.1.16.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	ArrayWindowDel (0x09B)	StartResult Ack	SenderHandle, FktIDArrayWindow, Tag, Quantity
		ErrorAck	SenderHandle, ErrorCode, ErrorInfo
		Processing Ack	SenderHandle
		ResultAck	SenderHandle

2.1.16.2 Parameter

SenderHandle

FktIDArrayWindow

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

Tag

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

Quantity

Number of rows.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

2.1.17 PowerDownDelay (0x0A0)

Section type: Coordination

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	Duration
		Get	-
		SetGet	Duration
		Status	Duration
		Error	ErrorCode, ErrorInfo

2.1.17.2 Parameter

Duration

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	min

2.1.18 SourceInfo (0x100)

Section type: Coordination

This property gives particulars about the type of streaming source data. If notification is used for an FBlock with multiple sources, multiple notifications will be sent when the status changes for more than one source.

2.1.18.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceInfo (0x100)	Get	SourceNr
		Status	SourceNr, DataType, DataDescription
		Error	ErrorCode, ErrorInfo

2.1.18.2 Parameter

SourceNr

Number of the data source.

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

DataType

Type of streaming data.

Basis datatype	Range of values	Code	Description
Enum	0x00..0xFF	0x00	PCM
		0x01	CDROM
		0x02	SPDIF
		0x20	MPEG1 System Stream
		0x21	MPEG2 Program Stream
		0x22	MPEG2 Transport Stream
		0x40	MPEG1 DTCP System Stream
		0x41	MPEG2 DTCP Program Stream
		0x42	MPEG2 DTCP Transport Stream
		0xFF	Unknown

DataDescription

Depending on DataType, additional information will be transported in DataDescription.

Basis datatype	Length	Condition	Description
Stream		DataType = 0x00	Content: Resolution, AudioChannels, SrcDelay, ChannelList
		DataType = 0x01	Content: BlockWidth, ChannelList
		DataType = 0x02	Content: ChannelList
		DataType = 0x20	Content: BlockWidth, ChannelList

		DataType = 0x21	Content: BlockWidth, ChannelList
		DataType = 0x22	Content: BlockWidth, ChannelList
		DataType = 0x40	Content: BlockWidth, ChannelList
		DataType = 0x41	Content: BlockWidth, ChannelList
		DataType = 0x42	Content: BlockWidth, ChannelList

Resolution

Resolution of the audio samples in bytes.

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

AudioChannels

Number of audio channels.

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

SrcDelay

Delay of streaming data related to the TimingMaster.

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

ChannelList

List of particular channels.

Basis datatype	Length	Condition	Description
Stream		-	Content: Channel Channel {, Channel}

Channel

Number of a channel.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..59	1	none

BlockWidth

Number of transferred bytes per MOST frame.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1..60	1	none

2.1.19 Allocate (0x101)

Section type: Coordination

The Allocate method causes the source to occupy Streaming Channels.

2.1.19.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	Allocate (0x101)	StartResult	SourceNr
		Processing	-
		Result	SourceNr, SrcDelay, ChannelList
		Error	ErrorCode, ErrorInfo

2.1.19.2 Parameter

SourceNr

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

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

SrcDelay

Delay of streaming data related to the TimingMaster.

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

ChannelList

List of particular channels.

Basis datatype	Length	Condition	Description
Stream		-	Content: Channel Channel {, Channel}

Channel

Number of a channel.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..59	1	none

2.1.20 DeAllocate (0x102)

Section type: Coordination

The method DeAllocate causes the source to free occupied Streaming Channels.

2.1.20.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DeAllocate (0x102)	StartResult	SourceNr
		Processing	-
		Result	SourceNr
		Error	ErrorCode, ErrorInfo

2.1.20.2 Parameter

SourceNr

Number of the data source (there can be several sources in one FBlock), for example, 0x01 for the first source.

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

2.1.21 SourceActivity (0x103)

Section type: Coordination

This method controls the activity of a source.

2.1.21.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceActivity (0x103)	StartResult	SourceNr, Activity
		Processing	-
		Result	SourceNr, Activity
		Error	ErrorCode, ErrorInfo

2.1.21.2 Parameter

SourceNr

Number of the data source.

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

Activity

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

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	Off
		0x01	Pause
		0x02	On

2.1.22 SourceName (0x104)

Section type: Coordination

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.

2.1.22.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceName (0x104)	Get	SourceNr
		Status	SourceNr, SourceName
		Error	ErrorCode, ErrorInfo

2.1.22.2 Parameter

SourceNr

Number of the data source.

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

SourceName

Name of the source.

Basis datatype	MaxSize
String	11

2.1.23 SourceConnect (0x105)

Section type: Coordination

By use of the method SourceConnect a source will connect their data to the given MOST Streaming Channels.

Note: In systems without a ConnectionMaster, the methods Allocate/Deallocate must be used to route streaming data to the MOST bus!

In systems with a ConnectionMaster, it is up to such master to decide whether allocation or source routing is used throughout the system.

2.1.23.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceConnect (0x105)	StartResult	SourceNr, ChannelList
		Processing	-
		Result	SourceNr, SrcDelay
		Error	ErrorCode, ErrorInfo

2.1.23.2 Parameter

SourceNr

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

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

ChannelList

List of particular channels.

Basis datatype	Length	Condition	Description
Stream		-	Content: Channel Channel {, Channel}

Channel

Number of a channel.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..59	1	none

SrcDelay

Delay of streaming data related to the TimingMaster.

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

2.1.24 SourceDisConnect (0x106)

Section type: Coordination

By use of the method SourceDisConnect the Streaming Channels of a source will be disconnected. This is for use with the method SourceConnect only.

2.1.24.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceDisConnect (0x106)	StartResult	SourceNr
		Processing	-
		Result	SourceNr
		Error	ErrorCode, ErrorInfo

2.1.24.2 Parameter

SourceNr

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

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

2.1.25 SourceRouting (0x107)

Section type: Coordination

This property describes the relation between the source numbers of the FBlock and the physically existing streaming data sources. Use this property to determine which source numbers are mutually exclusive.

2.1.25.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
GeneralFBlock	SourceRouting (0x107)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.25.2 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 property has only one dimension, y is unused.

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

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

Data

The content depends on the parameter Pos.

Basis datatype	Length	Description	
Array	-	Pos	Data
		{ x=0 }	PhysicalSource[1], PhysicalSource[2],...,PhysicalSource[NMax]
		{ x>0 }	PhysicalSource[x]

PhysicalSource

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

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

2.1.26 SinkInfo (0x110)

Section type: Coordination

The property SinkInfo can be used to query the sink about the type of data it can handle. If notification is used for an FBlock with multiple sinks, multiple notifications will be sent when the status changes for more then one sink.

2.1.26.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SinkInfo (0x110)	Get	SinkNr
		Status	SinkNr, DataType, DataDescription
		Error	ErrorCode, ErrorInfo

2.1.26.2 Parameter

SinkNr

Number of a data sink.

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

DataType

Type of the streaming data stream.

Basis datatype	Range of values	Code	Description
Enum	0x00..0xFF	0x00	PCM
		0x01	CDROM
		0x02	SPDIF
		0x20	MPEG1 System Stream
		0x21	MPEG2 Program Stream
		0x22	MPEG2 Transport Stream
		0x40	MPEG1 DTCP System Stream
		0x41	MPEG2 DTCP Program Stream
		0x42	MPEG2 DTCP Transport Stream
		0xFF	Unknown

DataDescription

Depending on DataType, additional information will be transported in DataDescription.

Basis datatype	Length	Condition	Description
Stream		DataType = 0x00	Content: Resolution, AudioChannels, SinkDelay, ChannelList
		DataType = 0x01	Content: BlockWidth, ChannelList
		DataType = 0x02	Content: ChannelList
		DataType = 0x20	Content: BlockWidth, ChannelList
		DataType = 0x21	Content: BlockWidth, ChannelList
		DataType = 0x22	Content: BlockWidth, ChannelList
		DataType = 0x40	Content: BlockWidth, ChannelList
		DataType = 0x41	Content: BlockWidth, ChannelList
		DataType = 0x42	Content: BlockWidth, ChannelList

Resolution

Resolution of the audio samples in bytes.

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

AudioChannels

Number of audio channels.

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

SinkDelay

Delay of streaming data related to the TimingMaster.

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

ChannelList

List of particular channels.

Basis datatype	Length	Condition	Description
Stream		-	Content: Channel Channel {, Channel}

Channel

Number of a channel.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..59	1	none

BlockWidth

Number of transferred bytes per MOST frame.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1..60	1	none

2.1.27 Connect (0x111)

Section type: Coordination

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

2.1.27.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	Connect (0x111)	StartResult	SinkNr, SrcDelay, ChannelList
		Processing	-
		Result	SinkNr
		Error	ErrorCode, ErrorInfo

2.1.27.2 Parameter

SinkNr

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

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

SrcDelay

Delay of streaming data related to the TimingMaster.

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

ChannelList

List of particular channels.

Basis datatype	Length	Condition	Description
Stream		-	Content: Channel Channel {, Channel}

Channel

Number of a channel.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..59	1	none

2.1.28 Disconnect (0x112)

Section type: Coordination

By use of the method Disconnect, channels for data streaming will be disconnected.

2.1.28.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	Disconnect (0x112)	StartResult	SinkNr
		Processing	-
		Result	SinkNr
		Error	ErrorCode, ErrorInfo

2.1.28.2 Parameter

SinkNr

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

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

2.1.29 Mute (0x113)

Section type: Coordination

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.

2.1.29.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	Mute (0x113)	Get	SinkNr
		SetGet	SinkNr, Status
		Status	SinkNr, Status
		Error	ErrorCode, ErrorInfo

2.1.29.2 Parameter

SinkNr

Number of 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 datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

Status

The Status parameter determines whether a sink is muted.

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

2.1.30 SinkName (0x114)

Section type: Coordination

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.

2.1.30.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SinkName (0x114)	Get	SinkNr
		Status	SinkNr, SinkName
		Error	ErrorCode, ErrorInfo

2.1.30.2 Parameter

SinkNr

Number of the streaming data sink.

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

SinkName

Name of the streaming data sink.

Basis datatype	MaxSize
String	11

2.1.31 ConnectTo (0x115)

Section type: Coordination

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

2.1.31.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	ConnectTo (0x115)	StartResult	FBlockID, InstID, SourceNr
		Processing	-
		Result	FBlockID, InstID, SourceNr
		Error	ErrorCode, ErrorInfo

2.1.31.2 Parameter

FBlockID

ID the FBlock that owns the source.

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

InstID

Instance ID of the FBlock that owns the source.

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

2.1.32 SyncDataInfo (0x116)

Section type: Coordination

This property SyncDataInfo can be used to query the FBlock on how many connections it may serve as sink or source. The usage of this function is optional.

2.1.32.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	SyncDataInfo (0x116)	Get	-
		Status	SourceCount, SinkCount
		Error	ErrorCode, ErrorInfo

2.1.32.2 Parameter

SourceCount

Number of possible source connections.

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

SinkCount

Number of possible sink connections.

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

2.1.33 SinkRouting (0x117)

Section type: Coordination

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.

2.1.33.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
GeneralFBlock	SinkRouting (0x117)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.33.2 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 property has only one dimension, y is unused. Valid range: x=1..number of sources (as given in SyncDataInfo), y=0

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

Data

The content depends on the parameter Pos.

Basis datatype	Length	Description	
Array	-	Pos	Data
		{ x=0 }	PhysicalSink [1], PhysicalSink [2], ..., 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 datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	none

2.1.34 DTCP_StartProcess (0x120)

Section type: Coordination

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.

2.1.34.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_Start Process (0x120)	Abort	-
		StartResult	FBlockID, InstID
		Processing	-
		Result	FBlockID, InstID
		Error	ErrorCode, ErrorInfo

2.1.34.2 Parameter

FBlockID

Functional address of the function block of the source

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

InstID

Instance ID of the function block of the source

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

2.1.35 DTCP_Control (0x121)

Section type: Coordination

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.

2.1.35.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_Control (0x121)	StartResult	RequesterFBlockID, RequesterInstID, Control_5C
		Processing	-
		Result	RequesterFBlockID, RequesterInstID, Control_5C
		Error	ErrorCode, ErrorInfo

2.1.35.2 Parameter

RequesterFBlockID

Functional address of the function block that sends a DTCP command.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

RequesterInstID

Instance ID of the function block that sends a DTCP command.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

Control_5C

The contents and structure of the Control fields are detailed in "Supplement B" of the 5C DTCP Specification

Basis datatype	Length	Description
Stream	-	

2.1.36 DTCP_Status (0x122)

Section type: Coordination

This function transmits DTCP status commands and corresponding responses.

2.1.36.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_Status (0x122)	StartResult	RequesterFBlockID, RequesterInstID, Status_5C
		Processing	-
		Result	RequesterFBlockID, RequesterInstID, Status_5C
		Error	ErrorCode, ErrorInfo

2.1.36.2 Parameter

RequesterFBlockID

Functional address of the function block that sends a DTCP command.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

RequesterInstID

Instance ID of the function block that sends a DTCP command.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

Status_5C

The contents and structure of the Status fields are detailed in "Supplement B" of the 5C DTCP Specification

Basis datatype	Length	Description
Stream		

2.1.37 DTCP_CipherStatus (0x123)

Section type: Coordination

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.

2.1.37.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_Cipher Status (0x123)	Get	SourceSinkNr
		Status	SourceSinkNr, AuthenticationState, AvailableExchangeKeys, CipherError
		Error	ErrorCode, ErrorInfo

2.1.37.2 Parameter

SourceSinkNr

Number of a data source or sink.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

AuthenticationState

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

Basis datatype	Range of values	Code	Description
Enum	0x00..0x05	0x00	State A0: Unauthenticated
		0x01	State A1: Full Authentication
		0x02	State A2: Restricted Authentication
		0x03	State A3: Authenticated
		0x04	State A4: Send Content Channel Key
		0x05	State A5: Initialize Device

AvailableExchangeKeys

AvailableExchangeKeys gives the current set of available ExchangeKeys

Basis datatype	Bit #	Code	Description
BitField	Bit 0	False	ExchangeKey for EMI Mode A (Copy-never) not available
		True	ExchangeKey for EMI Mode A (Copy-never) available

	Bit 1	False	ExchangeKey for EMI Mode B (Copy-one-generation) not available
		True	ExchangeKey for EMI Mode B (Copy-one-generation) available
	Bit 2	False	ExchangeKey for EMI Mode C (No-more-copies) not available
		True	ExchangeKey for EMI Mode C (No-more-copies) available
	Bit 3	False	Reserved
		True	Reserved
	Bit 4	False	DTCP-IP ExchangeKey for all E-EMI modes not available
		True	DTCP-IP ExchangeKey for all E-EMI modes available
	Bit 5..7	False	Reserved
		True	Reserved

CipherError

CipherError gives the current state of the ciphering machines

Basis datatype	Range of values	Code	Description
Enum	0x00..0x20	0x00	No error
		0x10	Encoding Error
		0x20	Decoding Error

2.1.38 DTCP_Info (0x124)

Section type: Coordination

This function gives information about MOST DTCP parameters.

2.1.38.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_Info (0x124)	Get	SourceSinkNr
		Status	SourceSinkNr, PacketFormat, MediaType, Packetlength, EncryptionFrameSize
		Error	ErrorCode, ErrorInfo

2.1.38.2 Parameter

SourceSinkNr

Number of a data source or sink.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

PacketFormat

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

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	Not defined
		0x01	Not applicable
		0x02	Generic MOST-DTCP Packet Format

MediaType

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

Packetlength

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

EncryptionFrameSize

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

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

2.1.39 DTCP_ContentKeyProcess (0x125)

Section type: Coordination

The method starts the establishing of Content Keys.

2.1.39.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
GeneralFBlock	DTCP_ContentKey Process (0x125)	StartResult	FBlockID, InstID, SourceNr, SinkNr, PacketFormat, MediaType, Packetlength, EncryptionFrameSize
		Processing	-
		Result	FBlockID, InstID
		Error	ErrorCode, ErrorInfo

2.1.39.2 Parameter

FBlockID

Functional address of the function block of the source.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

InstID

Instance ID of the function block instance of the source.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

SourceNr

Number of data source

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

SinkNr

Number of data sink

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

PacketFormat

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

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	Not defined
		0x01	Not applicable
		0x02	Generic MOST-DTCP Packet Format

MediaType

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	full range	1	none

Packetlength

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	full range	1	none

EncryptionFrameSize

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

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