

# MOST

Media Oriented Systems Transport

Multimedia and Control  
Networking Technology

**MOST FunctionBlock MicrophoneInput**

**Rev 2.3.1**

**09/2003**



## Legal Notice

### COPYRIGHT

© Copyright 1999 - 2003 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](mailto:contact@mostcooperation.com)

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



© Copyright 1999 - 2003 MOST Cooperation  
All rights reserved

MOST is a registered trademark

<b>1</b>	<b>INTRODUCTION .....</b>	<b>7</b>
<b>2</b>	<b>FUNCTIONBLOCK DEFINITION .....</b>	<b>7</b>
2.1	MicrophoneInput (FBlockID=0x26) .....	7
2.1.1	FktIDs (0x000) .....	7
2.1.2	Notification (0x001) .....	7
2.1.3	NotificationCheck (0x002) .....	9
2.1.4	SourceInfo (0x100) .....	10
2.1.5	Allocate (0x101) .....	12
2.1.6	DeAllocate (0x102) .....	13
2.1.7	SourceActivity (0x103) .....	13
2.1.8	SourceName (0x104) .....	14
2.1.9	SourceConnect (0x105) .....	14
2.1.10	SourceDisConnect (0x106) .....	16
2.1.11	SourceRouting (0x107) .....	16
2.1.12	SyncDataInfo (0x116) .....	17
<b>3</b>	<b>FUNCTIONBLOCK DYNAMIC SPECIFICATION .....</b>	<b>18</b>

## Bibliography MOST Function Catalog

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

FBlockID	FunctionBlock
0x00	GeneralFBlock
0x00	GeneralPlayer
0x01	NetBlock
0x02	NetworkMaster
0x03	ConnectionMaster
0x06	Diagnosis
0x0F	Enhanced Testability
0x22	AudioAmplifier
0x26	MicrophoneInput
0x30	AudioTapePlayer
0x31	AudioDiskPlayer
0x34	DVDVideoPlayer
0x40	AmFmTuner
0x41	TMCTuner
0x42	TVTuner
0x50	Telephone
0x51	GeneralPhoneBook
0x60	GraphicDisplay
0xFF	UniqueFunctions

## MicrophoneInput FBlock (0x26) Change History

Changes MicrophoneInput FBlock 2.3 to MicrophoneInput FBlock 2.3.1

Change Ref.	FktID	Changes
2.3.1-001	0x002	- Changed description of parameter FktIDList.
		-
		-

# 1 Introduction

A MOST Function Catalog is a collection of MOST FunctionBlocks.

This document contains the specification of a FunctionBlock. MOST FunctionBlocks 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 FunctionBlock Definition

### 2.1 MicrophoneInput (FBlockID=0x26)

Under certain system conditions it can be helpfull if a device can check whether its entries are still existent in the notification matrix or not. In case of error, a device is able to renew its entries.

#### 2.1.1 FktIDs (0x000)

With the property FktIDs the functions of a function block may be inquired.

##### 2.1.1.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	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	Description
Stream		FktID1, FktID2, ...

##### 2.1.2 Notification (0x001)

This property administrates the Notification Matrix of a function block.

### 2.1.2.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	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

Rx/TxLog of a device or group address

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

FktID

Function

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

DeviceIDList

List of Devices

Basis datatype	Length	Description
Stream		DeviceID {, DeviceID}



FktIDList

---

List of functions with a maximum of 4.

Basis datatype	Length	Description
Stream	8	FktID {, FktID}

## 2.1.3 NotificationCheck (0x002)

Under certain system conditions it can be helpfull if a device can check whether its entries are still existent in the notification matrix or not. In case of error, a device is able to renew its entries.

### 2.1.3.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	NotificationCheck (0x002)	Get	DeviceID
		Status	DeviceID, FktIDList
		Error	ErrorCode, ErrorInfo

### 2.1.3.2 Parameter

FktIDList

---

List of functions.

Basis datatype	Length	Description
Stream	-	FktID {, FktID}

FktID

---

Function

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

DeviceID

---

Rx/TxLog of a device or groupaddress

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

ErrorCode

---

ErrorInfo

---

## 2.1.4 SourceInfo (0x100)

This property gives particulars about the type of synchronous source data.

### 2.1.4.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	SourceInfo (0x100)	Get	SourceNr
		Status	SourceNr, DataType, DataDescription
		Error	ErrorCode, ErrorInfo

### 2.1.4.2 Parameter

DataType

---

Type of synchronous 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	Description	
Stream	-	DataType	Description
		0x00	Resolution, AudioChannels, SrcDelay, ChannelList
		0x01	Blockwidth, ChannelList
		0x02	ChannelList
		0x20	Blockwidth, ChannelList

		0x21	Blockwidth, ChannelList
		0x22	Blockwidth, ChannelList
		0x40	Blockwidth, ChannelList
		0x41	Blockwidth, ChannelList
		0x42	Blockwidth, ChannelList

## Resolution

Resolution of the AudioSamples in byte.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

## AudioChannels

Number of audio channels.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

## SrcDelay

Delay of synchronous Ddata related to the Timing Master. Remark: The parameter SrcDelay represents the register NDR.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

## ChannelList

List of particular channels.

Basis datatype	Length	Description
Stream	60	Channel {, Channel}

## Channel

Number of a channel

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..59	1	not_defined

## BlockWidth

Number of transferred byte per MOST frame.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

SourceNr

---

Number of data source.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

## 2.1.5 Allocate (0x101)

With this method Allocate the source will be caused to occupy synchronous channels.

### 2.1.5.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	Allocate (0x101)	Processing	
		Result	SourceNr, SrcDelay, ChannelList
		StartResult	SourceNr
		Error	ErrorCode, ErrorInfo

### 2.1.5.2 Parameter

SourceNr

---

Number of data source (within one function block there can be more than one), e.g. 0x01 for the first source.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

SrcDelay

---

Delay of synchronous data related to the Timing Master. Remark: The parameter SrcDelay represents the register NDR.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

ChannelList

---

List of particular Channels.

Basis datatype	Length	Description
Stream	60	Channel {, Channel}

Channel

Number of a channel

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..59	1	not_defined

## 2.1.6 DeAllocate (0x102)

The method DeAllocate causes the source to free occupied synchronous channels.

### 2.1.6.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	DeAllocate (0x102)	Processing	
		Result	SourceNr
		StartResult	SourceNr
		Error	ErrorCode, ErrorInfo

### 2.1.6.2 Parameter

SourceNr

Number of the data source (there can be several sources in one function block), e.g. 0x01 for the first source

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

## 2.1.7 SourceActivity (0x103)

This method controls the activity of an audio source.

### 2.1.7.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	SourceActivity (0x103)	Processing	
		Result	SourceNr, Activity
		StartResult	SourceNr, Activity
		Error	ErrorCode, ErrorInfo

## 2.1.7.2 Parameter

Activity

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

SourceNr

Number of data source.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

## 2.1.8 SourceName (0x104)

By property SourceName, an identifier of the synchronous source data can be requested.

### 2.1.8.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	SourceName (0x104)	Get	SourceNr
		Status	SourceNr, SourceName
		Error	ErrorCode, ErrorInfo

### 2.1.8.2 Parameter

SourceName

Basis datatype	MaxSize
String	11

SourceNr

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

## 2.1.9 SourceConnect (0x105)

By use of the method SourceConnect a source will connect their data to the given synchronous MOST channels. NOTE: In systems without a connection master, the methods

Allocate/Deallocate must be used to route synchronous data to the MOST bus! In systems with a connection master, it is up to such master to decide whether allocation or source routing is used throughout the system.

### 2.1.9.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	SourceConnect (0x105)	Processing	
		Result	SourceNr, SrcDelay
		StartResult	SourceNr, ChannelList
		Error	ErrorCode, ErrorInfo

### 2.1.9.2 Parameter

SourceNr

Number of data source (within one function block there can be more than one), e.g. 0x01 for the first source.

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

SrcDelay

Delay of synchronous data related to the Timing Master. Remark: The parameter SrcDelay represents the register NDR.

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

ChannelList

List of particular Channels.

Basis datatype	Length	Description
Stream	60	Channel {, Channel}

Channel

Number of a channel

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

## 2.1.10 SourceDisConnect (0x106)

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

### 2.1.10.1 Format of Function

**Function classes:** Unclassified Method

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	SourceDisConnect (0x106)	Processing	
		Result	SourceNr
		StartResult	SourceNr
		Error	ErrorCode, ErrorInfo

### 2.1.10.2 Parameter

SourceNr

Number of data source (within one function block there can be more than one), e.g. 0x01 for the first source.

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

## 2.1.11 SourceRouting (0x107)

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

### 2.1.11.1 Format of Function

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

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	SourceRouting (0x107)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

### 2.1.11.2 Parameter

Pos



The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. Since this property has only one dimension, y is unused. Valid range: x=1..number of sources (like 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	
Stream	-	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 connection master.

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

## 2.1.12 SyncDataInfo (0x116)

This property SyncDataInfo can be used to query the function block on how many connections it may serve as sink or source.

### 2.1.12.1 Format of Function

**Function classes:** Unclassified Property

FBlock	Function	OPType	Parameter
MicrophoneInput (0x26)	SyncDataInfo (0x116)	Get	
		Status	SourceCount, SinkCount
		Error	ErrorCode, ErrorInfo

### 2.1.12.2 Parameter

SinkCount

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

SourceCount

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0		1	not_defined

### 3 FunctionBlock Dynamic Specification



