MOST

Media **O**riented **S**ystems **T**ransport

Multimedia and Control Networking Technology

MOST FunctionBlock Audioamplifier
Rev 2.4.3
07/2011

Legal Notice

COPYRIGHT

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
D-76185 Karlsruhe
Germany

Tel: (+49) (0) 721 966 50 00
E-mail: contact@mostcooperation.com
Web: www.mostcooperation.com

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	2.5
[2]	MOST FBlock template GeneralFBlock	2.5.1

AudioAmplifier FBlock (0x22) Change History

Changes AudioAmplifier FBlock 2.4.2 to AudioAmplifier FBlock 2.4.3

Change Ref.	FktID	Changes
2.4.3-001	-	– Removed section that lists the currently released FBlocks. – Modified Introduction to match other MOST FBlock specifications. – Added bibliography section. – Removed empty Dynamic Specification chapter.
2.4.3-002	-	Removed "reserved" bits for Boolean data types.
2.4.3-003	-	Replaced "unit not defined" with unit "none".
2.4.3-004	0x000	Removed function FktIDs, which is now referenced from the GeneralFBlock.
2.4.3-005	0x001	Removed function Notification , which is now referenced from the GeneralFBlock.
2.4.3-006	0x002	Removed function NotificationCheck , which is now referenced from the GeneralFBlock.
2.4.3-007	0x110	Removed function SinkInfo , which is now referenced from the GeneralFBlock.
2.4.3-008	0x111	Removed function Connect , which is now referenced from the GeneralFBlock.
2.4.3-009	0x112	Removed function DisConnect , which is now referenced from the GeneralFBlock.
2.4.3-010	0x113	Removed function Mute , which is now referenced from the GeneralFBlock.
2.4.3-011	0x114	Removed function SinkName , which is now referenced from the GeneralFBlock.
2.4.3-012	0x115	Removed function ConnectTo , which is now referenced from the GeneralFBlock.
2.4.3-013	0x116	Removed function SyncDataInfo , which is now referenced from the GeneralFBlock.
2.4.3-014	0x117	Removed function SinkRouting , which is now referenced from the GeneralFBlock.
2.4.3-015	0x200	Corrected description of NSteps parameter.
2.4.3-015	-	Replaced function section entries with Occurrence attribute.
2.4.3-016	-	Added symbolic names for all Enum data types.
2.4.3-017	0x469	Changed BitField to BoolField of Unsigned Word.
2.4.3-018	0x46D	SpeakerLevel: Changed layout of function into Array of Array; it was incorrectly modeled as Array of Number.

Changes AudioAmplifier FBlock 2.4.1 to AudioAmplifier FBlock 2.4.2

Change Ref.	FktID	Changes
2.4.2-001	0x202	- Changed description of parameter NSteps.
2.4.2-002	0x400	- Changed description of parameter NSteps. - Changed range of parameter Volume.
2.4.2-003	0x440	- Changed description of parameter Pos.
2.4.2-004	0x441	- Changed description of parameter Pos.

Changes AudioAmplifier FBlock 2.4 to AudioAmplifier FBlock 2.4.1

Change Ref.	FktID	Changes
2.4.1-001	0x002	- Changed description of parameter FktIDList.
		-
		-

1 Introduction

This document contains the specification of an FBlock. 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 Function Catalog

2.1 AudioAmplifier (FBlockID=0x22)

This function block makes the functions for Amplifier applications available.

In addition to the functions contained in this document, the following functions are also part of this FBlock. They exist in the GeneralFBlock template and are included here by reference:

FktID	Function name
0x000	FktIDs
0x001	Notification
0x002	NotificationCheck
0x010	Version
0x110	SinkInfo
0x111	Connect
0x112	DisConnect
0x113	Mute
0x114	SinkName
0x115	ConnectTo
0x116	SyncDataInfo
0x117	SinkRouting

Function Overview		
FktID	**Name**	**Occurrence**
0x200	Balance	Mandatory
0x201	Loudness	Mandatory
0x202	Bass	Mandatory
0x203	Treble	Mandatory
0x204	Fader	Mandatory
0x400	Volume	Optional
0x401	FadeInOut	Optional
0x402	Subwoofer	Optional
0x404	BassBoost	Optional
0x421	CompThreshold	Optional
0x422	LimThreshold	Optional
0x423	CompGain	Optional
0x424	AttackTime	Optional
0x425	ReleaseTime	Optional
0x426	CompressorSettings	Optional
0x427	LimiterSettings	Optional
0x430	Crossover	Optional
0x431	CrossoverSlope	Optional
0x440	DelayLine	Optional
0x441	SpeakerDelay	Optional
0x450	InputGainOffset	Optional
0x451	OutputGainOffset	Optional
0x452	OutputPhase	Optional
0x460	EqualizerOnOff	Optional
0x461	EqualizerSettings	Optional
0x462	GraphEqualizerOnOff	Optional
0x463	GraphEqualizer	Optional
0x464	GraphEqualizerLinear	Optional
0x465	MidTones	Optional
0x466	MuteParameters	Optional
0x467	MixerLevel	Optional
0x468	SoundSettingList	Optional
0x469	RecallSoundSetting	Optional
0x46A	SaveSoundSetting	Optional
0x46B	DynSoundControl	Optional
0x46C	CurrentSoundSetting	Optional
0x46D	SpeakerLevel	Optional

2.1.1 Balance (0x200)

Occurrence: Mandatory

The balance adjusts the sound pattern between the left and right sink group. The adjustment is done by respective left/right symmetric volume commands (Balance-commands).

2.1.1.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	Balance (0x200)	Set	Balance
		Get	-
		SetGet	Balance
		Increment	NSteps
		Decrement	NSteps
		Status	Balance
		Error	ErrorCode, ErrorInfo

2.1.1.2 Parameter

Balance

Explicit values ranges are specified in the application. Negative values result in sound shifted to the right. That means, with a minimum value sound is only audible to the right. Analogous positive values result in sound shifted to the left. That means, with a maximum value sound is only audible to the left.

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

NSteps

Explicit values ranges are specified in the application.

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

2.1.2 Loudness (0x201)

Occurrence: Mandatory

To compensate the human level dependent audio sensitivity low frequencies will be boosted at low volume (loudness function). Switching it on/off is only mentioned for test- and bug-purposes. This function is not accessible to the customer.

2.1.2.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	Loudness (0x201)	Set	LoudnessOnOff
		Get	-
		SetGet	LoudnessOnOff
		Status	LoudnessOnOff
		Error	ErrorCode, ErrorInfo

2.1.2.2 Parameter

LoudnessOnOff

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

2.1.3 Bass (0x202)

Occurrence: Mandatory

To individually control the sound of the entertainment source there are separate bass- and treble-adjustments assigned.

2.1.3.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	Bass (0x202)	Set	Bass
		Get	-
		SetGet	Bass
		Increment	NSteps
		Decrement	NSteps
		Status	Bass
		Error	ErrorCode, ErrorInfo

2.1.3.2 Parameter

Bass

Explicit values ranges are specified in the application.

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

NSteps

Explicit values ranges are specified in the application.

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

2.1.4 Treble (0x203)

Occurrence: Mandatory

To individually control the sound of the entertainment source there are separate bass- and treble-adjustments assigned.

2.1.4.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	Treble (0x203)	Set	Treble
		Get	-
		SetGet	Treble
		Increment	NSteps
		Decrement	NSteps
		Status	Treble
		Error	ErrorCode, ErrorInfo

2.1.4.2 Parameter

Treble

Explicit values ranges are specified in the application.

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

NSteps

Explicit values ranges are specified in the application.

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

2.1.5 Fader (0x204)

Occurrence: Mandatory

The Fader adjusts the sound pattern between the front and rear sink group. The adjustment is done by respective left/right symmetric volume commands.

2.1.5.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	Fader (0x204)	Set	Fader
		Get	-
		SetGet	Fader
		Increment	NSteps
		Decrement	NSteps
		Status	Fader
		Error	ErrorCode, ErrorInfo

2.1.5.2 Parameter

Fader

Explicit values ranges are specified in the application. Negative values result in sound shifted to the front. That means, with a minimum value sound is only audible to the front. Analogous positive values result in sound shifted backwards. That means, with a maximum value sound is only audible in the back.

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

NSteps

Explicit values ranges are specified in the application.

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

2.1.6 Volume (0x400)

Occurrence: Optional

The volume can be set or read by use of this property.

2.1.6.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	Volume (0x400)	Set	Volume
		Get	-
		SetGet	Volume
		Increment	NSteps
		Decrement	NSteps
		Status	Volume
		Error	ErrorCode, ErrorInfo

2.1.6.2 Parameter

Volume

Explicit values ranges are specified in the application. The value 0 corresponds with mute. high values correspond with high volume.

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

NSteps

Explicit values ranges are specified in the application.

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

2.1.7 FadeInOut (0x401)

Occurrence: Optional

Comfortable Mute/Demute Note: This function is obsolete use Mute property (FktID=0x113) with SinkNr=0 instead.

2.1.7.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	FadeInOut (0x401)	Set	OnOff
		Get	-
		SetGet	OnOff
		Status	OnOff
		Error	ErrorCode, ErrorInfo

2.1.7.2 Parameter

OnOff

Basis data type	Bit #	Code	Description
Boolean	Bit 0	True	On, muted
		False	Off, un-muted

2.1.8 Subwoofer (0x402)

Occurrence: Optional

To individually control the sound of the entertainment source with seperate Subwoofer-adjustment.

2.1.8.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	Subwoofer (0x402)	Set	Subwoofer
		Get	-
		SetGet	Subwoofer
		Increment	NSteps
		Decrement	NSteps
		Status	Subwoofer
		Error	ErrorCode, ErrorInfo

2.1.8.2 Parameter

Subwoofer

Explicit values ranges are specified in the application.

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

NSteps

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

2.1.9 BassBoost (0x404)

Occurrence: Optional

This poperty controls the bass boost function, typically in a headphone amplifier.

2.1.9.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	BassBoost (0x404)	Set	OnOff
		Get	-
		SetGet	OnOff
		Status	OnOff
		Error	ErrorCode, ErrorInfo

2.1.9.2 Parameter

OnOff

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

2.1.10 CompThreshold (0x421)

Occurrence: Optional

The Compressor Threshold is the boundary (in dB), where compression of the input signal starts (an input signal with great dynamic range is mapped onto an output signal with small dynamic range). The compressor threshold always has to be smaller than the Limiter Threshold.

2.1.10.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	CompThreshold (0x421)	Set	CompThreshold
		Get	-
		SetGet	CompThreshold
		Status	CompThreshold
		Error	ErrorCode, ErrorInfo

2.1.10.2 Parameter

CompThreshold

Explicit values ranges are specified in the application. The value 0 corresponds with mute. high values correspond with high volume.

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

2.1.11 LimThreshold (0x422)

Occurrence: Optional

The Limiter Threshold (in dB) limits the amplitude of the output signal to the adjusted value. It always has to above the compressor threshold.

2.1.11.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	LimThreshold (0x422)	Set	LimThreshold
		Get	-
		SetGet	LimThreshold
		Status	LimThreshold
		Error	ErrorCode, ErrorInfo

2.1.11.2 Parameter

LimThreshold

Explicit values ranges are specified in the application. The value 0 corresponds with mute. high values correspond with high volume.

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

2.1.12 CompGain (0x423)

Occurrence: Optional

The Compressor Gain shows the amplification of the output signal towards the input signal in dB. Input signals of low level uniformly will be boosted by this value until the compressor threshold is reached.

2.1.12.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	CompGain (0x423)	Set	CompGain
		Get	-
		SetGet	CompGain
		Status	CompGain
		Error	ErrorCode, ErrorInfo

2.1.12.2 Parameter

CompGain

Explicit values ranges are specified in the application. The value 0 corresponds with mute. high values correspond with high volume.

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

2.1.13 AttackTime (0x424)

Occurrence: Optional

Attack Time is the rise time or response time of the compressor.

2.1.13.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	AttackTime (0x424)	Set	AttackTime
		Get	-
		SetGet	AttackTime
		Status	AttackTime
		Error	ErrorCode, ErrorInfo

2.1.13.2 Parameter

AttackTime

Explicit values ranges are specified in the application. The value 0 corresponds with mute. high values correspond with high volume.

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

2.1.14 ReleaseTime (0x425)

Occurrence: Optional

Release Time is the damping constant of the compression filter.

2.1.14.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	ReleaseTime (0x425)	Set	ReleaseTime
		Get	-
		SetGet	ReleaseTime
		Status	ReleaseTime
		Error	ErrorCode, ErrorInfo

2.1.14.2 Parameter

ReleaseTime

Explicit values ranges are specified in the application. The value 0 corresponds with mute. high values correspond with high volume.

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

2.1.15 CompressorSettings (0x426)

Occurrence: Optional

This function sets the compressor parameters.

2.1.15.1 Format of Function

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

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	CompressorSettings (0x426)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.15.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. The byte x represents one compressor band. The number NMax depends on the system. The byte y corresponds to compressor parameter.

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	
Array	-	Pos	Data
		{ x=0, y=0 }	{Frequency[1], Gain[1], Qvalue[1], AttackTime[1], AttackThreshold[1], ReleaseTime[1], ReleaseThreshold[1],...,Frequency[NMax], Gain[NMax], Qvalue[NMax], AttackTime[NMax], AttackThreshold[NMax], ReleaseTime[NMax], ReleaseThreshold[NMax]}

Frequency

Center frequency where the compressor band band work.

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0	20...20000	1	Hz

Gain

The gain shows the amplification towards the input signal in dB

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

Qvalue

The quality factor of the compressor band

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

AttackTime

The rise or respond time of the compressor

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

AttackThreshold

The boundary where compression of input signal starts

Basis data type	Exp.	Range of values	Step	Unit
Signed Byte	0	-127...0	1	dB

ReleaseTime

The damping constant of the compression filter

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

ReleaseThreshold

The boundary where compression of input signal stops

Basis data type	Exp.	Range of values	Step	Unit
Signed Byte	0	-127...0	1	dB

2.1.16 LimiterSettings (0x427)

Occurrence: Optional

This function sets the limiter parameters

2.1.16.1 Format of Function

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

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	LimiterSettings (0x427)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.16.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. The byte x represents one compressor band. The number NMax depends on the system. The byte y corresponds to compressor parameter

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

Data

Basis data type	Length	Description	
Array	-	Pos	Data
		{ x=0, y=0 }	{Threshold[1], AttackTime[1], ReleaseTime[1],...,Threshold[NMax], AttackTime[NMax], ReleaseTime[NMax]}

Threshold

The boundary where limiting of input signal starts

Basis data type	Exp.	Range of values	Step	Unit
Signed Byte	0	-127...0	1	dB

AttackTime

The rise or response time of the limiter

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

ReleaseTime

The damping constant of the limiter

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

2.1.17 Crossover (0x430)

Occurrence: Optional

The frequency switch is built up from the properties Low and High Frequency. They contain the 3-dB cut-off-frequency of the filter. Whereas you need both of them at a bandpass filter (combination of high- and lowpass filter), for a lowpass filter only the parameter XOverHiFrq (XOverLoFrq = 0) and for a highpass filter only XOverLoFrq (XOverHiFrq = 0) has to be set. For the setting Flat both parameters have to be 0. For each of the 6 processed audio-channels there's one crossover available. The crossover slope of the filters is 12dB/octave. The frequency range extends from 40Hz to 15kHz.

2.1.17.1 Format of Function

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

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	Crossover (0x430)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.17.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. Valid range: x=0..NMax, y=0..2

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	
Array	-	Pos	Data
		{ x=0, y=0 }	{XOverLoFreq[1], XOverHiFreq[1],...,XOverLoFreq[NMax], XOverHiFreq[NMax]}

XOverLoFreq

3-dB cut-off frequency for highpass filter

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

XOverHiFreq

3-dB cut-off frequency for lowpass filter

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

2.1.18 CrossoverSlope (0x431)

Occurrence: Optional

The slope (ordinal number) of the crossover filters may be adjusted by this property.

2.1.18.1 Format of Function

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

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	CrossoverSlope (0x431)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.18.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. Valid range: x=0..NMax, y=0..2

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	
Array	-	Pos	Data
		{ x=0, y=0 }	{XOverLoOrd[1], XOverHiOrd[1],...,XOverLoOrd[NMax], XOverHiOrd[NMax]}

XOverLoOrd

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...4	1	none

XOverHiOrd

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1...4	1	none

2.1.19 DelayLine (0x440)

Occurrence: Optional

This function makes it possible to do runtime corrections on particular speaker sets.

2.1.19.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	DelayLine (0x440)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.19.2 Parameter

Pos

Pos={x,y} consists of two byte, x and y, and indicates which parameter shall be set or read. The byte x identifies the number of the sink (SinkNr). Since this is an unidimensional construction, the second byte y is unused and the simplified notation Pos{x} is valid. Pos = 0 allows to set or get the parameters of all sinks. Valid range: x=0..NMax, y=0. NMax depends on the system.

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	
Array	-	Pos	Data
		{ x=0 }	{DelayTime[1],...,DelayTime[NMax]}
		{ x> 0 }	{DelayTime[x]}

DelayTime

Basis data type	Exp.	Range of values	Step	Unit
Unsigned Word	0	0...10000	1	ms

2.1.20 SpeakerDelay (0x441)

Occurrence: Optional

This function makes it possible to use different output channel delays for different speakers.

2.1.20.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	SpeakerDelay (0x441)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.20.2 Parameter

Pos

Pos={x,y} consists of two byte, x and y, and indicates which parameter shall be set or read. The byte x corresponds to speaker output. Since this is an unidimensional construction, the second byte y is unused and the simplified notation Pos{x} is valid. Pos = 0 allows to set or get the delay for all output channels. The number XNMax depends on the system.

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

Data

Basis data type	Length	Description	
Array	-	Pos	Data
		{ x=0 }	{Delay[1],...,Delay[6]}
		{ x> 0 }	{Delay[x]}

Delay

This parameter sets the delay for each speaker

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

2.1.21 InputGainOffset (0x450)

Occurrence: Optional

This property is intended to adjust level differences of different audio sources. There is one Gain Offset assigned for each audio input. In this function, the parameter Pos identifies the number of the sink (SinkNr) that the delay parameters are specified for. Pos = 0 allows to set or get the parameters of all sinks.

2.1.21.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	InputGainOffset (0x450)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.21.2 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 is an unidimensional construction, the second Byte y is unused (y=0=const) and the simplified notation Pos={x} is valid. Valid range: x=0..6, y=0

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

Data

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

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

Offset

Gain/Attenuation in dB

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

2.1.22 OutputGainOffset (0x451)

Occurrence: Optional

The property Output Gain Offset controls the attenuation devices in the output paths/channels. It is intended for adjusting speakers of different efficiency. Hints: The parameter Pos represents the SinkNr.

2.1.22.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	OutputGainOffset (0x451)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.22.2 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 is an unidimensional construction, the second Byte y is unused (y=0=const) and the simplified notation Pos={x} is valid. The byte x represents in this case the OutputNo. Valid range: x=0..Nmax, y=0

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

Data

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

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

Offset

Gain/Attenuation in dB

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

2.1.23 OutputPhase (0x452)

Occurrence: Optional

Phase of output signal

2.1.23.1 Format of Function

Function classes: Array of { Boolean }

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	OutputPhase (0x452)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.23.2 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 is an unidimensional construction, the second Byte y is unused (y=0=const) and the simplified notation Pos={x} is valid. Valid range: x=0..NMax with NMax, y=0

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	
Array	-	Pos	Data
		{ x=0 }	{OutputPhase[1],...,OutputPhase[2]}
		{ x> 0 }	{OutputPhase[x]}

OutputPhase

Basis data type	Bit #	Code	Description
Boolean	Bit 0	True	180°
		False	0°

2.1.24 EqualizerOnOff (0x460)

Occurrence: Optional

Phase of output signal

2.1.24.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	EqualizerOnOff (0x460)	Set	EquOnOff
		Get	-
		SetGet	EquOnOff
		Status	EquOnOff
		Error	ErrorCode, ErrorInfo

2.1.24.2 Parameter

EquOnOff

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

2.1.25 EqualizerSettings (0x461)

Occurrence: Optional

The property EqualizerSettings shows the amplification or attenuation respectively. It also shows the center frequency and the quality of the equalizer stage. Hints: The parameter Pos represents the SinkNr.

2.1.25.1 Format of Function

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

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	EqualizerSettings (0x461)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.25.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. The byte x represents the equalizer stage. The number of selectable stages Nmax depends on the equalizer. Valid range: x=0..NMax, y=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	
Array	-	Pos	Data
		{ x=0, y=0 }	{EquGain[1], EquFrequency[1], EquQuality[1],...,EquGain[NMax], EquFrequency[NMax], EquQuality[NMax]}

EquGain

Gain or attenuation

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

EquFrequency

Center frequency

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

EquQuality

Quality

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

2.1.26 GraphEqualizerOnOff (0x462)

Occurrence: Optional

The Audio Master Application only provides the interface for the graphical equalizer. Hint: Reserved for development purposes.

2.1.26.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	GraphEqualizerOnOff (0x462)	Set	GraphEquOnOff
		Get	-
		SetGet	GraphEquOnOff
		Status	GraphEquOnOff
		Error	ErrorCode, ErrorInfo

2.1.26.2 Parameter

GraphEquOnOff

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

2.1.27 GraphEqualizer (0x463)

Occurrence: Optional

The property GraphicalEqualizer shows the gain, or attenuation respectively, of the equalizer stage related to the center frequency. Datatype: Array [1..NMax] of {Gain} with NMax = number of stages Hints: The parameter Pos represents the SinkNr.

2.1.27.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	GraphEqualizer (0x463)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.27.2 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 is an unidimensional construction, the second Byte y is unused (y=0=const) and the simplified notation Pos={x} is valid. Valid range: x=0..NMax with NMax=7, y=0

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

Data

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

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

Gain

Gain or attenuation

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

2.1.28 GraphEqualizerLinear (0x464)

Occurrence: Optional

This method switches the graphical equalizer (all 7 channels) to linear.

2.1.28.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	GraphEqualizer Linear (0x464)	Start	-
		StartResult	-
		Processing	-
		Result	-
		Error	ErrorCode, ErrorInfo

2.1.29 MidTones (0x465)

Occurrence: Optional

This property contains the gain value for the middle frequency filter (Bass,MidTones,Treble).

2.1.29.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	MidTones (0x465)	Set	MidTones
		Get	-
		Increment	NSteps
		Decrement	NSteps
		Status	MidTones
		Error	ErrorCode, ErrorInfo

2.1.29.2 Parameter

MidTones

this is the gain the middle frequency filter. Explicit values ranges are specified in the application.

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

NSteps

No Description

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

2.1.30 MuteParameters (0x466)

Occurrence: Optional

This property allows to specify the characteristics of the mute process.

2.1.30.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	MuteParameters (0x466)	Set	SinkNr, MuteTimeIn, MuteTimeOut, MuteShapeIn, MuteShapeOut
		Get	SinkNr
		Status	SinkNr, MuteTimeIn, MuteTimeOut, MuteShapeIn, MuteShapeOut
		Error	ErrorCode, ErrorInfo

2.1.30.2 Parameter

SinkNr

Number of the sink

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

MuteTimeIn

Time to mute the sink

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

MuteTimeOut

Time to de-mute the sink

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

MuteShapeIn

Volume shape for the mute process

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x03	0x00	jump	jump
		0x01	linear	linear
		0x02	degressive	degressive
		0x03	progressive	progressive

MuteShapeOut

Volume shape for the de-mute process

Basis data type	Range of values	Code	Symbolic Name	Description
Enum	0x00...0x03	0x00	jump	jump
		0x01	linear	linear
		0x02	degressive	degressive
		0x03	progressive	progressive

2.1.31 MixerLevel (0x467)

Occurrence: Optional

This property allows to adjust gain, balance and fader of the various sinks of an amplifier. Hints: The parameter Pos represents the SinkNr.

2.1.31.1 Format of Function

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

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	MixerLevel (0x467)	Set	Pos, Data
		Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.31.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. Valid range: x=0..NMax, y=0

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	
Array	-	Pos	Data
		{ x=0, y=0 }	{InputLevel[1], Balance[1], Fader[1],...,InputLevel[NMax], Balance[NMax], Fader[NMax]}

InputLevel

contains the gain of the associated sink

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

Balance

adjusts balance of stereo sinks or panning of mono sinks 0x80 means that the sink balance is synchronized with the main balance

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

Fader

adjusts fading for the associated sink. 0x80 means that the sink fader is synchronized with the main fader

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

2.1.32 SoundSettingList (0x468)

Occurrence: Optional

This list contains the names of the available sound settings and the information whether the setting can be changed

2.1.32.1 Format of Function

Function classes: Array of { Record of { Boolean String } }

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	SoundSettingList (0x468)	Set	Pos, Data
		Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.32.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. Valid range: x=0..NMax, y=0

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	
Array	-	Pos	Data
		{ x=0, y=0 }	{ReadOnly[1], Name[1],...,ReadOnly[NMax], Name[NMax]}

ReadOnly

This value is true if the setting cannot be changed

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

Name

Name of the sound setting

Basis data type	MaxSize
String	#NULL#

2.1.33 RecallSoundSetting (0x469)

Occurrence: Optional

This method allows to recall a sound setting.

2.1.33.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	RecallSound Setting (0x469)	StartResult	Index, Categories
		Result	Index, Categories
		Error	ErrorCode, ErrorInfo

2.1.33.2 Parameter

Index

Index of the entry in the list of soundsettings which shall be recalled.

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

Categories

controls which setings shall be recalled from the memory.

Basis data type	Bit #	Code	Description
Unsigned Word	Bit 0	False	EQ Settings not changed
		True	EQ Settings changed
	Bit 1	False	Room SIM Settings not changed
		True	Room SIM Settings changed

2.1.34 SaveSoundSetting (0x46A)

Occurrence: Optional

This method allows to recall a saved sound setting.

2.1.34.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	SaveSoundSetting (0x46A)	StartResult	Index, Name
		Result	Index, Name
		Error	ErrorCode, ErrorInfo

2.1.34.2 Parameter

Index

Index where the current settings shall be stored in the list of sound settings. Contains 0 if the settings could not be stored.

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

Name

name of the sound setting

Basis data type	MaxSize
String	#NULL#

2.1.35 DynSoundControl (0x46B)

Occurrence: Optional

This property activates or deactivates all dynamic volume or sound adjustments (for example speed dependant volume adjustment)

2.1.35.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	DynSoundControl (0x46B)	Set	OnOff
		Get	-
		Status	OnOff
		Error	ErrorCode, ErrorInfo

2.1.35.2 Parameter

OnOff

This value is true if the dynamic volume or sound adjustment is active

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

2.1.36 CurrentSoundSetting (0x46C)

Occurrence: Optional

This property allows to specify the current sound setting.

2.1.36.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	CurrentSoundSetting (0x46C)	Set	SoundSetting
		Get	-
		SetGet	SoundSetting
		Increment	NSteps
		Decrement	NSteps
		Status	SoundSetting
		Error	ErrorCode, ErrorInfo

2.1.36.2 Parameter

SoundSetting

Index where the current settings shall be stored in the list of sound settings. Contains 0 if the settings could not be stored.

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

NSteps

Index where the current settings shall be stored in the list of sound settings. Contains 0 if the settings could not be stored.

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

2.1.37 SpeakerLevel (0x46D)

Occurrence: Optional

This function sets the level sound for each speaker.

2.1.37.1 Format of Function

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

FBlock	Function	OPType	Parameter
AudioAmplifier (0x22)	SpeakerLevel (0x46D)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.37.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read. The byte x represents the sink. The byte y corresponds to compressor parameter. The number XNMax and YNMax depends on the system.

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

Data

Basis data type	Length	Description	
Array	-	Pos	Data
		{ x=0, y=0 }	{Level(Data[1],LevelY[1]), Level(Data[1],LevelY[2]),...,Level(Data[x],LevelY[y]),...,Level(Data[XNMax],LevelY[YNMax])}
		{ x> 0, y=0 }	{Level(Data[x],LevelY[1]), Level(Data[x],LevelY[2]),...,Level(Data[x],LevelY[YNMax])}
		{ x> 0, y> 0 }	{Level(Data[x],LevelY[y])}

LevelY

Array of compressor parameters.

Basis data type	Element type	Element name
Array	Signed Byte	Level

Level

0 dB means maximum volume. -x dB means an attenuation of x dB relative to maximum volume. -127 dB means mute.

Basis data type	Exp.	Range of values	Step	Unit
Signed Byte	0	-127...0	1	dB