

MOST

Media Oriented Systems Transport

Multimedia and Control
Networking Technology

MOST FunctionBlock Telephone

Rev 2.3.2

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

Web: www.mostcooperation.com



© Copyright 1999 - 2003 MOST Cooperation
All rights reserved

MOST is a registered trademark

1	INTRODUCTION	8
2	FUNCTIONBLOCK DEFINITION	8
2.1	Telephone (FBlockID=0x50)	8
2.1.1	FktIDs (0x000)	8
2.1.2	Notification (0x001)	8
2.1.3	NotificationCheck (0x002)	10
2.1.4	SourceInfo (0x100)	11
2.1.5	Allocate (0x101)	13
2.1.6	DeAllocate (0x102)	14
2.1.7	SourceActivity (0x103)	14
2.1.8	SourceName (0x104)	15
2.1.9	SourceConnect (0x105)	15
2.1.10	SourceDisConnect (0x106)	17
2.1.11	SourceRouting (0x107)	17
2.1.12	SinkInfo (0x110)	18
2.1.13	Connect (0x111)	20
2.1.14	DisConnect (0x112)	21
2.1.15	Mute (0x113)	22
2.1.16	SinkName (0x114)	22
2.1.17	ConnectTo (0x115)	23
2.1.18	SyncDataInfo (0x116)	24
2.1.19	SinkRouting (0x117)	24
2.1.20	PhoneEnable (0x200)	25
2.1.21	ChangeCode (0x201)	26
2.1.22	LockState (0x202)	27
2.1.23	DialNumber (0x250)	28
2.1.24	HangupCall (0x251)	28
2.1.25	CurrentNumber (0x252)	29
2.1.26	CallState (0x253)	29
2.1.27	CallDuration (0x254)	31
2.1.28	AcceptCall (0x255)	32
2.1.29	CallInfo (0x25A)	32
2.1.30	CallHold (0x260)	33
2.1.31	ResumeCall (0x261)	34
2.1.32	CallForward (0x262)	34
2.1.33	CallDeflection (0x263)	36
2.1.34	CallBarring (0x264)	36
2.1.35	FollowUpTime (0x400)	38
2.1.36	RestoreFactorySettings (0x401)	38
2.1.37	CLIR (0x402)	39
2.1.38	CLIP (0x403)	40
2.1.39	CLOP (0x404)	41
2.1.40	NetworkOperatorSelection (0x420)	41
2.1.41	NetworkOperators (0x421)	42
2.1.42	HomeNetwork (0x422)	44
2.1.43	SignalQuality (0x423)	44
2.1.44	SerialNumber (0x424)	45
2.1.45	MobileStandard (0x425)	46
2.1.46	LineState (0x426)	47
2.1.47	RegisterState (0x427)	48
2.1.48	PressedKey (0x428)	48
2.1.49	VoiceMailboxNumber (0x441)	49
2.1.50	VoiceMailboxConnect (0x442)	49
2.1.51	SendDTMF (0x443)	50
2.1.52	HandsFreeOnOff (0x446)	50
2.1.53	SimCardReaderInfo (0x462)	51
2.1.54	PowerSupply (0x466)	52

2.1.55	PhoneDate (0x490)	53
2.1.56	PhoneTime (0x491)	54
2.1.57	AdviceOfCharge (0x492)	55
2.1.58	SMSShowList (0x500)	56
2.1.59	SMSShowDetails (0x501)	58
2.1.60	SMSSend (0x502)	59
2.1.61	SMSSendFromStorage (0x503)	61
2.1.62	SMSSStore (0x504)	63
2.1.63	SMSNew (0x505)	63
2.1.64	SMSAttrib (0x506)	65
2.1.65	SMSDelete (0x507)	65
2.1.66	SMSCBClear (0x510)	66
2.1.67	SMSCBReceive (0x511)	67
2.1.68	SMSCBSettings (0x512)	68
2.1.69	SMSSStorages (0x513)	70
2.1.70	SMSSStorageSelect (0x514)	71
2.1.71	SMSBinaryShowList (0x520)	72
2.1.72	SMSBinaryShowDetails (0x521)	74
2.1.73	SMSBinarySend (0x522)	75
2.1.74	MPMultiPartyCall (0x550)	76
2.1.75	MPReleaseActiveCallAcceptHeldCall (0x551)	77
2.1.76	MPReleaseActiveCallAcceptWaitingCall (0x552)	77
2.1.77	MPSwap (0x553)	78
2.1.78	MPCallHoldAcceptWaitingCall (0x554)	78
2.1.79	MPReleaseAllCallsExceptWaitingCall (0x555)	79
2.1.80	MPReleaseAllCallsAcceptWaitingCall (0x556)	79
2.1.81	MPExplicitCallTransfer (0x557)	79
2.1.82	CCConferenceCall (0x558)	80
2.1.83	CCSplit (0x559)	81
2.1.84	CCJoin (0x55A)	81
2.1.85	WaitingCallAlert (0x560)	82
2.1.86	OpenDataConnection (0x600)	82
2.1.87	EndDataConnection (0x601)	84
2.1.88	AcceptDataConnection (0x602)	85
2.1.89	DataCapabilities (0x603)	85
2.1.90	ConnectionStatus (0x604)	87
2.1.91	DataSpeedCapabilities (0x605)	89
3	FUNCTIONBLOCK DYNAMIC SPECIFICATION	90

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

Document History

Changes MOST TelephoneFBlock 2V3-01 to MOST TelephoneFBlock 2V3-02

Change Ref.	FktID	Changes
2.3.2-001	0x002	- Changed description of parameter FktIDList.
2.3.2-002	0x424	- Fixed error in SIMSerial parameter. Data type has been changed to String.
		-

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 Telephone (FBlockID=0x50)

Telephone contains the necessary functions to establish a mobile interconnection.

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
Telephone (0x50)	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
Telephone (0x50)	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
Telephone (0x50)	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
Telephone (0x50)	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
Telephone (0x50)	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
Telephone (0x50)	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 methode controls the activity of an audio source.

2.1.7.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	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
Telephone (0x50)	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
Telephone (0x50)	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
Telephone (0x50)	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
Telephone (0x50)	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 SinkInfo (0x110)

The property SinkInfo can be used to query the sink about the type of data it can handle.

2.1.12.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
Telephone (0x50)	SinkInfo (0x110)	Get	SinkNr
		Status	SinkNr, DataType, DataDescription
		Error	ErrorCode, ErrorInfo

2.1.12.2 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	DataDescription
		0x00	{Resolution, AudioChannels, SrcDelay, Channels}
		0x01	{Blockwidth, Channels}
		0x02	{ChannelList}
		0x20	{Blockwidth, Channels}
		0x21	{Blockwidth, Channels}
		0x22	{Blockwidth, Channels}
		0x40	{Blockwidth, Channels}
		0x41	{Blockwidth, Channels}
		0x42	{Blockwidth, Channels}

Resolution

Resolution of the AudioSamples in byte.

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

BlockWidth

Number of transferred byte per MOST frame.

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

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

SinkDelay

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	none

SinkNr

Number of a data sink.

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

2.1.13 Connect (0x111)

By use of the method Connect synchronous channels for audio reception will be connected.

2.1.13.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	Connect (0x111)	Processing	
		Result	SinkNr
		StartResult	SinkNr, SrcDelay, ChannelList
		Error	ErrorCode, ErrorInfo

2.1.13.2 Parameter

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

SinkNr

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

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

2.1.14 DisConnect (0x112)

By use of the method DisConnect synchronous channels for audio reception will be disconnected.

2.1.14.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	DisConnect (0x112)	Processing	
		Result	SinkNr
		StartResult	SinkNr
		Error	ErrorCode, ErrorInfo

2.1.14.2 Parameter

SinkNr

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

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

2.1.15 Mute (0x113)

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

2.1.15.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
Telephone (0x50)	Mute (0x113)	Get	SinkNr
		SetGet	SinkNr, Status
		Status	SinkNr, Status
		Error	ErrorCode, ErrorInfo

2.1.15.2 Parameter

SinkNr

Number of data sink (within one function block there can be more than one), e.g. 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

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

2.1.16 SinkName (0x114)

By using property SinkName, a name for the synchronous data can be requested.

2.1.16.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
Telephone (0x50)	SinkName (0x114)	Get	SinkNr
		Status	SinkNr, SinkName
		Error	ErrorCode, ErrorInfo

2.1.16.2 Parameter

SinkName

Basis datatype	MaxSize
String	11

SinkNr

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

2.1.17 ConnectTo (0x115)

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

2.1.17.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	ConnectTo (0x115)	Processing	
		Result	FBlockID, InstID, SourceNr
		StartResult	FBlockID, InstID, SourceNr
		Error	ErrorCode, ErrorInfo

2.1.17.2 Parameter

SourceNr

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

InstID

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

FBlockID

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

2.1.18 SyncDataInfo (0x116)

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

2.1.18.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
Telephone (0x50)	SyncDataInfo (0x116)	Get	
		Status	SourceCount, SinkCount
		Error	ErrorCode, ErrorInfo

2.1.18.2 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

2.1.19 SinkRouting (0x117)

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

2.1.19.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
Telephone (0x50)	SinkRouting (0x117)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.19.2 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 }	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 connection master.

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

2.1.20 PhoneEnable (0x200)

Input code to enable phone functionality.

2.1.20.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	PhoneEnable (0x200)	Processing	
		Result	
		Start	CodeType, Code
		StartResult	CodeType, Code
		Error	ErrorCode, ErrorInfo

2.1.20.2 Parameter

CodeType

Codetype to unlock the phone

Basis datatype	Range of values	Code	Description
Enum	0x00..0x07	0x00	Undef
		0x01	PIN
		0x02	PIN2
		0x03	PUK
		0x04	PUK2
		0x05	LockCode
		0x06	KeypadLockcode
		0x07	BarringPassword

Code

Code to unlock the phone

Basis datatype	MaxSize
String	20

2.1.21 ChangeCode (0x201)

Changes the lock code

2.1.21.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	ChangeCode (0x201)	Processing	
		Result	
		Start	CodeType, NewCode, CurrentCode
		StartResult	CodeType, NewCode, CurrentCode
		Error	ErrorCode, ErrorInfo

2.1.21.2 Parameter

CodeType

Codetype to unlock the phone

Basis datatype	Range of values	Code	Description
Enum	0x00..0x05	0x00	Undef
		0x01	PIN

		0x02	PIN2
		0x03	Lockcode
		0x04	KeypadLockcode
		0x05	BarringPassword

NewCode

New Code

Basis datatype	MaxSize
String	20

CurrentCode

Old Code

Basis datatype	MaxSize
String	20

2.1.22 LockState (0x202)

Status of phone lock

2.1.22.1 Format of Function

Function classes: Enumeration

FBlock	Function	OPType	Parameter
Telephone (0x50)	LockState (0x202)	Get	
		Status	LockState
		Error	ErrorCode, ErrorInfo

2.1.22.2 Parameter

LockState

Basis datatype	Range of values	Code	Description
Enum	0x00..0x0A	0x00	NoLock
		0x01	RequirePIN
		0x02	RequirePIN2
		0x03	PINblockedRequirePUK
		0x04	PIN2blockedRequirePUK2
		0x05	PUKblocked
		0x06	PUK2blocked

		0x07	KeypadLocked
		0x08	SIMNotAvailable
		0x09	PINinvalid
		0x0A	PIN2invalid

2.1.23 DialNumber (0x250)

This function dials the previously input phone number (empty string => redial). see ITU-T E164, too. DialNumber is not hanging up current established calls automatically.

2.1.23.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	DialNumber (0x250)	Processing	
		Result	CallId
		Start	TelNumber
		StartResult	TelNumber
		Error	ErrorCode, ErrorInfo

2.1.23.2 Parameter

CallId

Valid range 1..NMAX

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

TelNumber

The number to dial (empty string => redial) possible characters 0..9,*,#,"",n,+

Basis datatype	MaxSize
String	40

2.1.24 HangupCall (0x251)

Finish one or all existing calls. Release a call from a conference call

2.1.24.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	HangupCall (0x251)	Processing	
		Result	
		Start	CallId
		StartResult	CallId
		Error	ErrorCode, ErrorInfo

2.1.24.2 Parameter

CallId

CallId=0xFF ends all calls Valid range: 1..NMAX

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

2.1.25 CurrentNumber (0x252)

Last dialed number.

2.1.25.1 Format of Function

Function classes: Text

FBlock	Function	OPType	Parameter
Telephone (0x50)	CurrentNumber (0x252)	Set	TelNumber
		Get	
		SetGet	TelNumber
		Status	TelNumber
		Error	ErrorCode, ErrorInfo

2.1.25.2 Parameter

TelNumber

Phone number

Basis datatype	MaxSize
String	40

2.1.26 CallState (0x253)

Main states of a call with CallId By use of CallId it is easy to implement multiparty calls If one state changes, the complete array is to be transfered.

2.1.26.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	CallState (0x253)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.26.2 Parameter

Pos

pos consists of tow bytes x is equal to CallId Valid range: x=0..NMax, y=0..2

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..65535	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	CallState[1], CallType[1], CallState[2], CallType[2], ..., CallState[NMAX], CallType[NMAX]
		{ x>0, y=0 }	CallState[x], CallType[x]
		{ x>0, y=1 }	CallState[x]
		{ x>0, y=2 }	CallType[x]

CallState

0x06: call state for AMPS, TDMA, CDMA; 0x07..0x0b: call states for PDC

Basis datatype	Range of values	Code	Description
Enum	0x00..0x0b	0x00	Idle
		0x01	Ringing
		0x02	Active
		0x03	Dialing
		0x04	Disconnecting
		0x05	On Hold
		0x06	Reestablish
		0x07	Calling (RBT)
		0x08	Busy

		0x09	Multiparty Call Ringing
		0x0a	Multiparty Call Active
		0x0b	Conference Call

CallType

Basis datatype	Range of values	Code	Description
Enum	0x00..0x01	0x00	Normal Call
		0x01	Emergency Call

2.1.27 CallDuration (0x254)

Duration of all currently available calls

2.1.27.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
Telephone (0x50)	CallDuration (0x254)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.27.2 Parameter

Pos

pos is equal to call id Valid range: x=0..NMax, y=0

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..65535	1	not_defined

Data

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

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	Time(1), ..., Time(NMAX)
		{ x>0 }	Time(x)

Time

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	s

2.1.28 AcceptCall (0x255)

Accept an incoming call.

2.1.28.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	AcceptCall (0x255)	Processing	
		Result	
		Start	CallId
		StartResult	CallId
		Error	ErrorCode, ErrorInfo

2.1.28.2 Parameter

CallId

Unique index to differentiate the connections on multiparty calls Valid range: 1..NMAX

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

2.1.29 CallInfo (0x25A)

Information about all calls

2.1.29.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	CallInfo (0x25A)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.29.2 Parameter

Pos

The parameter Pos={x} [0..NMax] consists of two byte and shows what is to be set, requested or read in the record. Pos is equal to CallID Valid range: x=0..NMAX, y=0..2

Basis datatype	Exp.	Range of values	Step	Unit
----------------	------	-----------------	------	------

Unsigned Word	0		1	not_defined
---------------	---	--	---	-------------

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	TelNumber[1], Name[1], TelNumber[2], Name[2], ..TelNumber[NMAX], Name[NMAX]
		{ x>0, y=0 }	TelNumber[x], Name[x]
		{ x>0, y=1 }	TelNumber[x]
		{ x>0, y=2 }	Name[x]

TelNumber

Number of the caller if available, otherwise transfer empty string

Basis datatype	MaxSize
String	40

Name

Name of caller if available, otherwise an empty string will be transferred

Basis datatype	MaxSize
String	50

2.1.30 CallHold (0x260)

Places an active call on hold

2.1.30.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	CallHold (0x260)	Processing	
		Result	
		Start	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.30.2 Parameter

2.1.31 ResumeCall (0x261)

Activate a held call

2.1.31.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	ResumeCall (0x261)	Processing	
		Result	
		Start	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.31.2 Parameter

2.1.32 CallForward (0x262)

Call forwarding allows for control of the call forwarding supplementary service. Registration, erasure, activation, deactivation, and status query are supported. Hints: If the CallType is not set to 0xFF, the function only applies to that CallType only if Mode=2 Result delivers back meaningful values if Mode=1,2-> PhoneNumber and Time is ignored if Mode=3->

2.1.32.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	CallForward (0x262)	Processing	
		Result	, CallType, Status, PhoneNumber, Time
		Start	Reason, Mode, CallType, PhoneNumber, Time
		StartResult	Reason, Mode, CallType, PhoneNumber, Time, ErrorCode
		Error	ErrorCode, ErrorInfo

2.1.32.2 Parameter

Status

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	not active
		True	active
	Bit 1 ... 7	-	reserved

Time

1...30 when "no reply" is enabled or queried, this gives the time in seconds to wait before call is forwarded, default value 20

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	1..30	1	s

CallType

Basis datatype	Range of values	Code	Description
Enum	0x00..0x08	0x00	voice (telephony)
		0x01	data (refers to all bearer services)
		0x02	fax (facsimile services)
		0x03	short message service
		0x04	data circuit sync
		0x05	data circuit async
		0x06	dedicated packet access
		0x07	dedicated PAD access
		0x08	unconditional (applies to all call types)

PhoneNumber

Phone number of forwarding address

Basis datatype	MaxSize
String	40

Mode

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	disable
		0x01	enable
		0x02	query status
		0x03	registration
		0x04	ensure

Reason

All call forwarding and all conditional call forwarding (refer GSM 02.30)

Basis datatype	Range of values	Code	Description
Enum	0x00..0x05	0x00	unconditional
		0x01	mobile busy
		0x02	no reply
		0x03	not reachable
		0x04	all call forwarding
		0x05	all conditional call forwarding

2.1.33 CallDeflection (0x263)

Call deflection refers to a service that causes an incoming alerting call to be forwarded to a specified number.

2.1.33.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	CallDeflection (0x263)	Processing	
		Result	
		Start	PhoneNumber
		StartResult	PhoneNumber
		Error	ErrorCode, ErrorInfo

2.1.33.2 Parameter

PhoneNumber

Phone number of forwarding address

Basis datatype	MaxSize
String	40

2.1.34 CallBarring (0x264)

Call Barring is used to lock or unlock calls or query current barring status. Password is normally needed to do such actions. Hints: If the CallType is not set to 0xFF, the function only applies to that CallType. Only if Mode=2 Result delivers back meaningful values. If Mode=0,1 -> Type is ignored

2.1.34.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	CallBarring (0x264)	Processing	
		Result	Status, CallType
		Start	Type, Mode, CallType, BarringPassword
		StartResult	Type, Mode, CallType, BarringPassword
		Error	ErrorCode, ErrorInfo

2.1.34.2 Parameter

Mode

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	lock
		0x01	unlock
		0x02	query

BarringPassword

Basis datatype	MaxSize
String	20

CallType

Basis datatype	Range of values	Code	Description
Enum	0x00..0x05	0x00	Voice
		0x01	Fax
		0x02	Data
		0x03	SMS
		0x04	unspecified
		0x05	Emergency

Status

Basis datatype	Bit #	Code	Description
Boolean	Bit 0 ... 7	-	reserved

Type

0 BAOC (Barr All Outgoing Calls) (refer GSM 02.88 [6] clause 1 1 BOIC (Barr Outgoing International Calls) (refer GSM 02.88 [6] clause 1 2 BOIC?exHC (Barr Outgoing International Calls expt to Home Country) (refer GSM 02.88 [6] clause 1 3 BAIC (Barr All Incoming Calls) (refer GSM 02.88 [6] clause 2 4 BIC?Roam (Barr Incoming Calls when Roaming outside the home country) (refer GSM 02.88 [6] clause 2 5 All barring services

(refer GSM 02.30 [19]) (applicable only for <mode>=0 6 All outgoing barring services (refer GSM 02.30 [19]) (applicable only for <mode>=0 7 All incoming barring services (refer GSM 02.30 [19]) (applicable only for <mode>=0

Basis datatype	Range of values	Code	Description
Enum	0x00..0x07	0x00	BAOC
		0x01	BOIC
		0x02	BOIC?exHC
		0x03	BAIC
		0x04	BIC?Roam
		0x05	All barring services
		0x06	All outgoing barring services
		0x07	All incoming services

2.1.35 FollowUpTime (0x400)

Adjust follow-up time of the phone.

2.1.35.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
Telephone (0x50)	FollowUpTime (0x400)	Set	Time
		Get	
		SetGet	Time
		Status	Time
		Error	ErrorCode, ErrorInfo

2.1.35.2 Parameter

Time

Follow-up time; valid range: 0..NMAX

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0		1	min

2.1.36 RestoreFactorySettings (0x401)

Reset all phone settings to factory settings

2.1.36.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	RestoreFactorySettings (0x401)	Processing	
		Result	
		Start	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.36.2 Parameter

2.1.37 CLIR (0x402)

Calling Line Identification Restriction

2.1.37.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	CLIR (0x402)	Processing	
		Result	OutgoingCall, NetStatus
		Start	OutgoingCall
		StartResult	OutgoingCall
		Error	ErrorCode, ErrorInfo

2.1.37.2 Parameter

NetStatus

parameter shwos the subscriber CLIR service status in the network

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	CLIR not provisioned
		0x01	CLIR provisioned in permanent mode
		0x02	unknown (e.g. no network, etc.)
		0x03	CLIR temporary mode prenentation restricted
		0x04	CLIR temporary mode prenentation allowed

OutgoingCall

Parameter sets the adjustment for outgoing calls

Basis datatype	Range of values	Code	Description
Enum	0x00..0x03	0x00	Presentation indicator used according to the subscription of the CLIR services
		0x01	CLIR invocation
		0x02	CLIR suppression
		0x03	undefined (read mode)

2.1.38 CLIP (0x403)

Calling Line Identification Presentation

2.1.38.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	CLIP (0x403)	Processing	
		Result	PresentationStatus, NetStatus
		Start	PresentationStatus
		StartResult	PresentationStatus
		Error	ErrorCode, ErrorInfo

2.1.38.2 Parameter

NetStatus

Parameter shows the subscriber CLIP service status in the network

Basis datatype	Range of values	Code	Description
Enum	0x00..0x01	0x00	CLIP not provisioned
		0x01	CLIP provisioned

PresentationStatus

Parameter sets/shows the result code presentation status in the TA

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	disable
		0x01	enable
		0x02	undefined (read mode)

2.1.39 CLOP (0x404)

Connected Line Identification Presentation

2.1.39.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	CLOP (0x404)	Processing	
		Result	PresentationStatus, NetStatus
		Start	PresentationStatus
		StartResult	PresentationStatus
		Error	ErrorCode, ErrorInfo

2.1.39.2 Parameter

NetStatus

Parameter shows the subscriber COLP service status in the network

Basis datatype	Range of values	Code	Description
Enum	0x00..0x01	0x00	CLOP not provisioned
		0x01	CLOP provisioned

PresentationStatus

Parameter sets/shows the result code presentation status in the TA

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	disable
		0x01	enable
		0x02	undefined (read mode)

2.1.40 NetworkOperatorSelection (0x420)

For roaming conditions it is necessary to switch on manual operator selection

2.1.40.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
Telephone (0x50)	NetworkOperatorSelection (0x420)	Set	OnOff
		Get	

		Status	OnOff
		Error	ErrorCode, ErrorInfo

2.1.40.2 Parameter

OnOff

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Automatic
		True	Manual
	Bit 1 ... 7	-	reserved

2.1.41 NetworkOperators (0x421)

This property administrates the list of network providers. All available providers may be displayed. The current provider may be set.

2.1.41.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	NetworkOperators (0x421)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.41.2 Parameter

Pos

The parameter Pos={x,y} consists of two byte and shows which parameter(s) shall be set, requested or read. Valid range: x=0..NMax, y=0..4

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

Data

Depended on parameter Pos Data consists of the following parameters:

Basis datatype	Length	Description	
Stream	-	Pos	Data

		{ x=0, y=0 }	Selected[1], NetCode[1], NetName[1], MobileStandardType[1], Selected[2], NetCode[2], NetName[2], MobileStandardType[2], ..., Selected[NMax], NetCode[NMax], NetName[NMax], MobileStandardType[NMAX]
		{ x>0, y=0 }	Selected[x], NetCode[x], NetName[x]
		{ x>0, y=1 }	Selected[x]
		{ x>0, y=2 }	NetCode[x]
		{ x>0, y=3 }	NetName[x]
		{ x>0, y=4 }	MobileStandardType[x]

Selected

Basis datatype	Range of values	Code	Description
Enum	0x00..0x03	0x00	unknown
		0x01	available
		0x02	selected
		0x03	forbidden

NetCode

Code of network provider (cannot be changed) The NetCode consists of the CountryCode and the NetworkCode

Basis datatype	MaxSize
String	7

NetName

Name of network provider (cannot be set)

Basis datatype	MaxSize
String	50

MobileStandardType

Basis datatype	Range of values	Code	Description
Enum	0x00..0x12	0x00	AMPS800
		0x01	CDMA800
		0x02	CDMA1800
		0x03	CDMA1900
		0x04	TDMA800
		0x05	TDMA1900

	0x06	GSM400
	0x07	GSM900
	0x08	GSM1800
	0x09	GSM1900
	0x10	PDC800
	0x11	PDC1500
	0x12	WCDMA

2.1.42 HomeNetwork (0x422)

This property administrates the list of network providers. All available providers may be displayed. The current provider may be set.

2.1.42.1 Format of Function

Function classes: Text

FBlock	Function	OPType	Parameter
Telephone (0x50)	HomeNetwork (0x422)	Get	
		Status	NetCode
		Error	ErrorCode, ErrorInfo

2.1.42.2 Parameter

NetCode

The NetCode consists of CounrtyCode and NetworkCode

Basis datatype	MaxSize
String	7

2.1.43 SignalQuality (0x423)

Information about signal quality (fieldstrength and bit error rate).

2.1.43.1 Format of Function

Function classes: Number

FBlock	Function	OPType	Parameter
Telephone (0x50)	SignalQuality (0x423)	Get	
		Status	Quality
		Error	ErrorCode, ErrorInfo

2.1.43.2 Parameter

Quality

Signal Quality 0% no reception 100% Best reception

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..99	1	none

2.1.44 SerialNumber (0x424)

Delivers the serial number of the telephone (e.g. IMEI in GSM, ISN in AMPS/TDMA/CDMA).

2.1.44.1 Format of Function

Function classes: Record of { String String }

FBlock	Function	OPType	Parameter
Telephone (0x50)	SerialNumber (0x424)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.44.2 Parameter

Pos

Valid range: x=0..2, y=0

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..2	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	SerialNumber, SimSerial
		{ x=1 }	SerialNumber
		{ x=2 }	SimSerial

SerialNumber

Serial number of the telephone

Basis datatype	MaxSize
String	40

SIMSerial

Serial number of the SIM SIMSerial = 0 ==> not available

Basis datatype	MaxSize
String	40

2.1.45 MobileStandard (0x425)

MobileStandard determines the selected Mobile Standard like GSM, AMPS etc.

2.1.45.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	MobileStandard (0x425)	Set	Pos, Data
		Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.45.2 Parameter

Pos

The parameter Pos consists of two byte and shows what is to be set, requested or read in the record. Valid range: x=0..NMax, y=0..2

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

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	Selected[1], MobileStandardType[1], ..., Selected[Nmax], MobileStandardType[Nmax]
		{ x>0, y=0 }	Selected[x], MobileStandardType[x]
		{ x>0, y=1 }	Selected[x]
		{ x>0, y=2 }	MobileStandard[x]

Selected

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	not selected
		True	selected
	Bit 1 ... 7	-	reserved

MobilStandardType

Basis datatype	Range of values	Code	Description
Enum	0x00..0x12	0x00	AMPS800
		0x01	CDMA800
		0x02	CDMA1800
		0x03	CDMA1900
		0x04	TDMA800
		0x05	TDMA1900
		0x06	GSM400
		0x07	GSM900
		0x08	GSM1800
		0x09	GSM1900
		0x10	PDC800
		0x11	PDC1500
		0x12	WCDMA

2.1.46 LineState (0x426)

Status of dialed connection

2.1.46.1 Format of Function

Function classes: Enumeration

FBlock	Function	OPType	Parameter
Telephone (0x50)	LineState (0x426)	Get	
		Status	State
		Error	ErrorCode, ErrorInfo

2.1.46.2 Parameter

State

Basis datatype	Range of values	Code	Description
Enum	0x00..0x03	0x00	LineFree
		0x01	NoLine

		0x02	ConnectedLine/Busy
		0x03	SystemBusy

2.1.47 RegisterState (0x427)

Status of registration in the network

2.1.47.1 Format of Function

Function classes: Enumeration

FBlock	Function	OPType	Parameter
Telephone (0x50)	RegisterState (0x427)	Get	
		Status	RegisterState
		Error	ErrorCode, ErrorInfo

2.1.47.2 Parameter

RegisterState

Basis datatype	Range of values	Code	Description
Enum	0x00..0x05	0x00	not registered and not searching
		0x01	registered
		0x02	not registered and searching
		0x03	registration denied
		0x04	registered and roaming
		0x05	registered and roaming alternative

2.1.48 PressedKey (0x428)

Inform the telephone about a pressed key to enable key clicks or DTMF tones.

2.1.48.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	PressedKey (0x428)	Result	
		StartResult	Key
		Error	ErrorCode, ErrorInfo

2.1.48.2 Parameter

Key

The pressed key according to GSM 07.07 "keypad control"

Basis datatype	MaxSize
String	1

2.1.49 VoiceMailboxNumber (0x441)

This property selects the number of the voice box.

2.1.49.1 Format of Function

Function classes: Text

FBlock	Function	OPType	Parameter
Telephone (0x50)	VoiceMailboxNumber (0x441)	Set	VoiceBoxNumber
		Get	
		SetGet	VoiceBoxNumber
		Status	VoiceBoxNumber
		Error	ErrorCode, ErrorInfo

2.1.49.2 Parameter

VoiceBoxNumber

Voice Box Number at the operator

Basis datatype	MaxSize
String	40

2.1.50 VoiceMailboxConnect (0x442)

Connect to voice mailbox.

2.1.50.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	VoiceMailboxConnect (0x442)	Processing	
		Result	CallID
		Start	CallID
		StartResult	CallID
		Error	ErrorCode, ErrorInfo

2.1.50.2 Parameter

CallID

valid range: 0..NMAX

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

2.1.51 SendDTMF (0x443)

This method sends a DTMF character (e.g. for voice box). Only useful when a connection is already established.

2.1.51.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SendDTMF (0x443)	Processing	
		Result	
		Start	DTMFSign
		StartResult	DTMFSign
		Error	ErrorCode, ErrorInfo

2.1.51.2 Parameter

DTMFSign

DTMF Sign valid range: 0..9,*,#A..D,M

Basis datatype	MaxSize
String	40

2.1.52 HandsFreeOnOff (0x446)

Activate/deactivate hands free

2.1.52.1 Format of Function

Function classes: Switch

FBlock	Function	OPType	Parameter
Telephone (0x50)	HandsFreeOnOff (0x446)	Set	OnOff
		Get	

		Status	OnOff
		Error	ErrorCode, ErrorInfo

2.1.52.2 Parameter

OnOff

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

2.1.53 SimCardReaderInfo (0x462)

Information about the occupancy of the SIM card readers whether SIM card is plugged in or not. The function serves as event on a SIM card change.

2.1.53.1 Format of Function

Function classes: Array of { Boolean }

FBlock	Function	OPType	Parameter
Telephone (0x50)	SimCardReaderInfo (0x462)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.53.2 Parameter

Pos

The parameter Pos={ x } [0..NMax] consists of two byte and shows what is to be set, requested or read in the record. Valid range: x=0..NMax, y=0

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

Data

Depended on parameter Pos Data consists of the following parameters:

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	SimCard(1), ...SimCard(NMax)
		{ x>0 }	SimCard(x)

SimCard

Sim card x plugged in

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Sim card x not plugged in
		True	Sim card 1 plugged in
	Bit 1 ... 7	-	reserved

2.1.54 PowerSupply (0x466)

Supply condition of the phone and charging condition of main- and backup-battery

2.1.54.1 Format of Function

Function classes: Record of { Number Number }

FBlock	Function	OPType	Parameter
Telephone (0x50)	PowerSupply (0x466)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.54.2 Parameter

Pos

Valid range: x=0..2, y=0

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..2	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	MainCharge, BackupCharge
		{ x=1 }	MainCharge
		{ x=2 }	BackupCharge

MainCharge

Charging condition of 1. Battery

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

BackupCharge

Charge condition of the backup battery BackupCharge = 0 ==> BatteryFault, no Battery connected

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

2.1.55 PhoneDate (0x490)

Request and display the system date of the mobile communication network

2.1.55.1 Format of Function

Function classes: Record of { Number Number Number }

FBlock	Function	OPType	Parameter
Telephone (0x50)	PhoneDate (0x490)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.55.2 Parameter

Pos

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..3	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	Day, Month, Year
		{ x=1 }	Year
		{ x=2 }	Month
		{ x=3 }	Day

Day

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..31	1	d

Month

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..12	1	mon

Year

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	1999	1	a

2.1.56 PhoneTime (0x491)

Time of the phone

2.1.56.1 Format of Function

Function classes: Record of { Number Number Number Enumeration }

FBlock	Function	OPType	Parameter
Telephone (0x50)	PhoneTime (0x491)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.56.2 Parameter

Pos

Valid range: x=0..4, y=0

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..4	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	Hour, Minute, Second, AmPm
		{ x=1 }	Hour
		{ x=2 }	Minute
		{ x=3 }	Second
		{ x=4 }	AmPm

Hour

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Byte	0	0..23	1	h

Minute

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

Second

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

AmPm

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	none
		0x01	Am
		0x02	Pm

2.1.57 AdviceOfCharge (0x492)

Charge value: current state, maximum charge value, price per unit

2.1.57.1 Format of Function

Function classes: Record of { Number Number Number String }

FBlock	Function	OPType	Parameter
Telephone (0x50)	AdviceOfCharge (0x492)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Decrement	
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.57.2 Parameter

Pos

Valid range: x=0..4, y=0

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..4	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	Account, Maximum, PerUnit, Currency
		{ x=1 }	Account
		{ x=2 }	Maximum
		{ x=3 }	PerUnit
		{ x=4 }	Currency

Account

Accumulated charge

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

Maximum

Maximum charge value

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

PerUnit

Charge per minute

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	-2		1	none

Currency

Account currency

Basis datatype	MaxSize
String	6

2.1.58 SMSShowList (0x500)

Show text mode SMS that are stored in the phone. The state of parameter SMSAttrib.Switch is not influenced by SMSShowList

2.1.58.1 Format of Function

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

FBlock	Function	OPType	Parameter
--------	----------	--------	-----------

Telephone (0x50)	SMSShowList (0x500)	Set	
		Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.58.2 Parameter

Pos

The parameter Pos={x} consists of two byte and shows what is to be set, requested or read in the record. With SMSShowList only x=0 and y=0 are useful, because the whole list is transferred anyways. Valid range x=0..NMax, y=0..5

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

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	Loc[1], SMSAttrib[1], SMSDate[1], SMSTime[1], SMSName[1], ..Loc[NMax], SMSAttrib[NMax], SMSDate[NMax], SMSTime[NMax], SMSName[NMax]
		{ x>0, y=0 }	Loc[x], SMSAttrib[x], SMSDate[x], SMSTime[x], SMSName[x]
		{ x>0, y=1 }	Loc[x]
		{ x>0, y=2 }	SMSAttrib[x]
		{ x>0, y=3 }	SMSDate[x]
		{ x>0, y=4 }	SMSTime[x]
		{ x>0, y=5 }	SMSName[x]

Loc

Location Valid range: 0..NMAX

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

SMSAttrib

all: all SMSs are transferred

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	not read
		0x01	read
		0x02	not sent
		0x03	sent
		0x04	all

SMSSDate

Basis datatype	MaxSize
String	10

SMSTime

Basis datatype	MaxSize
String	7

SMSName

Basis datatype	MaxSize
String	40

2.1.59 SMSShowDetails (0x501)

SMS message

2.1.59.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSShowDetails (0x501)	Processing	
		Result	Loc, SMSCNumber, SMSName, TelNumber, SMSText, ErrorCode, SMSSDate, SMSTime, SMSAttrib
		Start	Loc
		StartResult	Loc
		Error	ErrorCode, ErrorInfo

2.1.59.2 Parameter

SMSAttrib

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	not read

		0x01	read
		0x02	not sent
		0x03	sent
		0x04	all

SMSTime

Basis datatype	MaxSize
String	7

SMSDate

Basis datatype	MaxSize
String	10

SMSText

Basis datatype	MaxSize
String	160

TelNumber

Basis datatype	MaxSize
String	40

SMSName

Basis datatype	MaxSize
String	50

SMSCNumber

Basis datatype	MaxSize
String	40

Loc

Location Valid range: 0..NMAX

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

2.1.60 SMSSend (0x502)

Send text SMS

2.1.60.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSSend (0x502)	Processing	
		Result	
		Start	TelNumber, SMSCNumber, SMSText, Storage, ReplyPath, ValidityPeriod, MsgConversion, RequestDeliveryReport
		StartResult	TelNumber, SMSCNumber, SMSText, Storage, ReplyPath, ValidityPeriod, MsgConversion, RequestDeliveryReport
		Error	ErrorCode, ErrorInfo

2.1.60.2 Parameter

RequestDeliveryReport

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Do not request report
		True	Request report
	Bit 1 ... 7	-	reserved

MsgConversion

Basis datatype	Range of values	Code	Description
Enum	0x00..0x05	0x00	none
		0x01	fax
		0x02	x400
		0x03	paging
		0x04	email
		0x05	speech

ValidityPeriod

Time period to fetch message. Otherwise the message will expire. 1 Byte = 00 relativ 8 Byte valid 1 Byte = 01 absolute time stamp yy mm dd hh ss timezone

Basis datatype	MaxSize
String	#NULL#

ReplyPath

Answer via same SMSCNumber

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	True, same SMSCNumber
		True	False, different SMSCNumber
	Bit 1 ... 7	-	reserved

Storage

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Do not store message
		True	Store message
	Bit 1 ... 7	-	reserved

SMSText

Basis datatype	MaxSize
String	160

SMSCNumber

Phone number of the SMS service center

Basis datatype	MaxSize
String	40

TelNumber

Basis datatype	MaxSize
String	40

2.1.61 SMSSendFromStorage (0x503)

Send SMS from SMS storage.

2.1.61.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSSendFromStorage (0x503)	Processing	
		Result	
		Start	TelNumber, SMSCNumber, ReplyPath, ValidityPeriod, MsgConversion, RequestDeliveryReport, Loc

		StartResult	TelNumber, SMSCNumber, ReplyPath, ValidityPeriod, MsgConversion, RequestDeliveryReport, Loc
		Error	ErrorCode, ErrorInfo

2.1.61.2 Parameter

Loc

Location Valid range: 0..NMAX

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

RequestDeliveryReport

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Do not request report
		True	Request report
	Bit 1 ... 7	-	reserved

MsgConversion

Basis datatype	Range of values	Code	Description
Enum	0x00..0x05	0x00	none
		0x01	fax
		0x02	x400
		0x03	paging
		0x04	email
		0x05	speech

ValidityPeriod

1 Byte = 00 relativ 8 Byte valid 1 Byte = 01 absolute time stamp yy mm dd hh ss timezone

Basis datatype	MaxSize
String	8

ReplyPath

Answer via same SMSCNumber

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	True, same SMSCNumber
		True	False, different SMSCNumber

	Bit 1 ... 7	-	reserved
--	-------------	---	----------

SMSCNumber

Phone number of the SMS service center

Basis datatype	MaxSize
String	40

TelNumber

Basis datatype	MaxSize
String	40

2.1.62 SMSStore (0x504)

Write SMS to SMS storage without sending it.

2.1.62.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSStore (0x504)	Processing	
		Result	Loc
		StartResult	SMSText
		Error	ErrorCode, ErrorInfo

2.1.62.2 Parameter

Loc

Location Valid range: 0..NMAX

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

SMSText

Basis datatype	MaxSize
String	160

2.1.63 SMSNew (0x505)

Display that new messages have arrived.

2.1.63.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSNew (0x505)	Get	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.63.2 Parameter

Pos

The parameter Pos={ x } consists of two byte and shows what is to be set, requested or read in the record. With SMSNew only x=0 and y=0 are useful, because the whole list is transferred anyways. Valid range x=0..NMax, y=0..2

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

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	Loc[1], ...Loc[NMax]
		{ x>0 }	Loc[x]
		{ x=0, y=0 }	Loc[1], Text_Binary[1], ...Loc[NMax], Text_Binary[NMax]
		{ x>0, y=0 }	Loc[x], Text_Binary[x]
		{ x>0, y=1 }	Loc[x]
		{ x>0, y=2 }	Text_Binary[x]

Loc

Location Valid range: 0..NMAX

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

Text_Binary

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	unknown
		0x01	Binary
		0x02	Text

2.1.64 SMSAttrib (0x506)

The parameter switch is only influenced by the Function SMSAttrib

2.1.64.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSAttrib (0x506)	Processing	
		Result	
		StartResult	Loc, Switch
		Error	ErrorCode, ErrorInfo

2.1.64.2 Parameter

Switch

Because SMS are displayed on MMIs that are not domiciled in the phone, the MMI must be able to actively mark an SMS as read.

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	unread
		True	read
	Bit 1 ... 7	-	reserved

Loc

Index on the SIM card; Valid range: 0..NMAX

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

2.1.65 SMSDelete (0x507)

Delete SMS in the SMS storage. Loc=FFFF -> Deletion of all SMS

2.1.65.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSDelete (0x507)	Processing	
		Result	
		Start	Loc

		StartResult	Loc
		Error	ErrorCode, ErrorInfo

2.1.65.2 Parameter

Loc

Location Valid range: 0..NMAX

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

2.1.66 SMSCBClear (0x510)

Delete, StartID=0, EndID=0xFFFF delete all

2.1.66.1 Format of Function

Function classes: Record of { Number Number Number }

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSCBClear (0x510)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.66.2 Parameter

Pos

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

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..3	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	DeviceID, StartID, EndID
		{ x=1 }	{DeviceID
		{ x=2 }	StartID
		{ x=3 }	EndID

EndID

End of range of booked messages

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

StartID

Start of range of booked messages

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

DeviceID

Address of MOST device to which the message is to be sent.

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

2.1.67 SMSCBReceive (0x511)

Reception of a CB message

2.1.67.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSCBReceive (0x511)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.67.2 Parameter

Pos

Valid range: x=0..5, y=0

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..5	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data

	{ x=0 }	SequenceNumber, MessageID, ParamDCS, Page, CBData
	{ x=1 }	SequenceNumber
	{ x=2 }	MessageID
	{ x=3 }	ParamDCS
	{ x=4 }	Page
	{ x=5 }	CBData

SequenceNumber

Internal label of the cellbroadcast message

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

MessageID

ID of booked message

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

ParamDCS

Format of booked message

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

Page

Internal label of the cellbroadcast message

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

CBData

Content of received message

Basis datatype	MaxSize
String	-

2.1.68 SMSCBSettings (0x512)

Booked service info scopes, e.g. stock exchange, traffic, weather, sports etc. Multiple subscriptions of services are possible, that means one service to several devices is possible as

well as several services to one device. The booked ID ranges will be deleted at shutdown of the system.

2.1.68.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSCBSettings (0x512)	Set	Pos, Data
		Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.68.2 Parameter

Pos

Valid range: x=0..NMax, y=0..4

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

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	DeviceID[1], StartID[1], EndID[1], ParamDCS[1], ..., DeviceID[NMax], StartID[NMax], EndID[NMax], ParamDCS[NMax]
		{ x=1, y=0 }	DeviceID[x], StartID[x], EndID[x], ParamDCS[x]
		{ x>0, y=1 }	DeviceID[x]
		{ x>0, y=2 }	StartID[x]
		{ x>0, y=3 }	EndID[x]
		{ x>0, y=4 }	ParamDCS[x]

DeviceID

Address of MOST device to which the message is to be sent.

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

StartID

Start of range of booked messages

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

EndID

End of range of booked messages

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

ParamDCS

Format of booked message

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

2.1.69 SMSStorages (0x513)

Retrieves the available SMS storages including information about used memory and total size of the storage. Remark: can also be notified.

2.1.69.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSStorages (0x513)	Get	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.69.2 Parameter

Pos

Pos consists of two bytes Valid range: x=0..NMax, y=0..3

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

Data

Dependent on parameter Pos, Data consists of the following parameters:

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	Storage[1], Used[1], Total[1], Storage[2], Used[2], Total[2], ..., Storage[NMax], Used[NMax], Total[NMax]
		{ x>0, y=0 }	Storage[x], Used[x], Total[x]
		{ x>0, y=1 }	Storage[x]
		{ x>0, y=2 }	Used[x]
		{ x>0, y=3 }	Total[x]

Storage

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	"ME" ME message storage
		0x01	"MT" any of the storages associated with ME
		0x02	"SM" SIM message storage
		0x03	"TA" TA message storage
		0x04	"NA" Not available

Used

Used indicates the number of used locations in selected memory

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

Total

Total indicates the total number of locations in selected memory

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

2.1.70 SMSStorageSelect (0x514)

Selection of different SMS storages

2.1.70.1 Format of Function

Function classes: Record of { Enumeration Enumeration }

FBlock	Function	OPType	Parameter
--------	----------	--------	-----------

Telephone (0x50)	SMSStorageSelect (0x514)	Set	Pos, Data
		Get	Pos
		SetGet	Pos, Data
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.70.2 Parameter

Pos

Valid range: x=0..2, y=0

Basis datatype	Exp.	Range of values	Step	Unit
Unsigned Word	0	0..2	1	not_defined

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	Type, Index
		{ x=1 }	Type
		{ x=2 }	Index

Type

Basis datatype	Range of values	Code	Description
Enum	0x00..0x01	0x00	Incoming SMS storage
		0x01	SMS storage to be operated (the selected storage)

Index

Index for an entry in a dedicated SMS storage list (see SMSStorages) 0..NMax Example:
Index Storage Used Total 1 ME xxx yyy 2 ME xxx yyy 3 TA xxx yyy

Basis datatype	Range of values	Code	Description
Enum	0x00..0x7F	0x00..0x7F	reserved

2.1.71 SMSBinaryShowList (0x520)

Show of all binary SMS. Mode=PDU; fix receive setting

2.1.71.1 Format of Function

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

FBlock	Function	OPType	Parameter
--------	----------	--------	-----------

Telephone (0x50)	SMSBinaryShowList (0x520)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.71.2 Parameter

Pos

The parameter Pos={x} consists of two byte and shows what is to be set, requested or read in the record. With SMSBinaryShowList only x=0 and y=0 are useful, because the whole list is transferred anyways. Valid range: x=0..NMax, y=0..4

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

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	Loc[1], SMSAttrib[1], SMSDate[1], SMSTime[1], ..Loc[NMax], SMSAttrib[NMax], SMSDate[NMax], SMSTime[NMax]
		{ x>0, y=0 }	Loc[x], SMSAttrib[x], SMSDate[x], SMSTime[x]
		{ x>0, y=1 }	Loc[x]
		{ x>0, y=2 }	SMSAttrib[x]
		{ x>0, y=3 }	SMSDate[x]
		{ x>0, y=4 }	SMSTime[x]

Loc

SIM Location Valid range: 0..NMAX

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

SMSAttrib

all: all SMSs are transferred

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	not read
		0x01	read

		0x02	not sent
		0x03	sent
		0x04	all

SMSTDate

Basis datatype	MaxSize
String	10

SMSTime

Basis datatype	MaxSize
String	7

2.1.72 SMSBinaryShowDetails (0x521)

Binary SMS

2.1.72.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSBinaryShowDetails (0x521)	Processing	
		Result	Loc, SMSCNumber, SMSName, TelNumber, SMSData
		Start	Loc
		StartResult	Loc
		Error	ErrorCode, ErrorInfo

2.1.72.2 Parameter

SMSTData

Received data without header

Basis datatype	Length	Descripton
Stream	-	Description

TelNumber

Basis datatype	MaxSize
String	40

SMSName

Basis datatype	MaxSize
String	50

SMSCNumber

Provider Telephone Number

Basis datatype	MaxSize
String	40

Loc

SIM Location Valid Range: 0..NMAX

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

2.1.73 SMSBinarySend (0x522)

Send binary SMS

2.1.73.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	SMSBinarySend (0x522)	Processing	
		Result	
		Start	SMSCNumber, SMSData
		StartResult	SMSCNumber, SMSData
		Error	ErrorCode, ErrorInfo

2.1.73.2 Parameter

SMSCNumber

phone number of SMS service center

Basis datatype	MaxSize
String	40

SMSData

All sent binary data including header according to GSM definition (GSM 03.40.)

Basis datatype	Length	Descripton
----------------	--------	------------

Stream	-	Description
--------	---	-------------

2.1.74 MPMultiPartyCall (0x550)

The MPMultiPartyCall property contains information about the current calls (existence and multiparty state). The Call-ID is a unique number and is used to reference a single call in the call array. The multiparty state describes the state of a dedicated call (Active, OnHold or Waiting). In all other call states the multiparty call is not existent in the MPMultiPartyCall property. In addition the exact call state can be resolved by using the CallState property (Idle, Dialing, Waiting/Ringing, Active, OnHold, Disconnecting).

2.1.74.1 Format of Function

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

FBlock	Function	OPType	Parameter
Telephone (0x50)	MPMultiPartyCall (0x550)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.74.2 Parameter

Pos

The parameter Pos consists of two byte and shows what is to be set, requested or read in the record. pos is equal to CallID Valid range: x=0..NMax, y=0..3

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

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0, y=0 }	Ready[1], Telnumber[1], State[1], Ready[2], Telnumber[2], State[2], ..., Ready[Nmax], Telnumber[Nmax], State[Nmax]
		{ x>0, y=0 }	Ready[x], Telnumber[x], State[x]
		{ x>0, y=1 }	Ready[x]
		{ x>0, y=2 }	Telnumber[x]
		{ x>0, y=3 }	State[x]

Ready

Call present

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	none present
		True	present
	Bit 1 ... 7	-	reserved

TelNumber

Phone number

Basis datatype	MaxSize
String	40

State

Call state (hold, active, waiting)

Basis datatype	Range of values	Code	Description
Enum	0x00..0x02	0x00	Hold
		0x01	Active
		0x02	Waiting

2.1.75 MPReleaseActiveCallAcceptHeldCall (0x551)

The active call is released and the held call becomes active. Both, the active or the held call, might be a conference. For this function, no Call-ID is required.

MPReleaseActiveCallAcceptHeldCall can not be used if there is neither an active nor a held call.

2.1.75.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	MPReleaseActiveCallAcceptHeldCall (0x551)	Result	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.75.2 Parameter

2.1.76 MPReleaseActiveCallAcceptWaitingCall (0x552)

The active call is released and the waiting call becomes active. The active call might be a conference. For this function, no Call-ID is required.

MPReleaseActiveCallAcceptWaitingCall can not be used if there is neither an active nor a waiting call.

2.1.76.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	MPReleaseActiveCallAcceptWaitingCall (0x552)	Result	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.76.2 Parameter

2.1.77 MPSwap (0x553)

The active call is put on hold and the held call becomes active. The active or the held call call might be a conference. For this function, no Call-ID is required. MPSwap can not be used if there is neither an active call nor a held call.

2.1.77.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	MPSwap (0x553)	Result	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.77.2 Parameter

2.1.78 MPCallHoldAcceptWaitingCall (0x554)

The active call is put on hold and the waiting call becomes active. The active might be a conference. For this function, no Call-ID is required. MPCallHoldAcceptWaitingCall can not be used if there is neither an active call nor a waiting call.

2.1.78.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	MPCallHoldAcceptWaitingCall (0x554)	Result	
		StartResult	

		Error	ErrorCode, ErrorInfo
--	--	-------	----------------------

2.1.78.2 Parameter

2.1.79 MPReleaseAllCallsExceptWaitingCall (0x555)

All calls (the active and the held) are released. The waiting call remains waiting. MPReleaseAllCallsExceptWaitingCall can not be used if there is no waiting call.

2.1.79.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	MPReleaseAllCallsExceptWaitingCall (0x555)	Result	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.79.2 Parameter

2.1.80 MPReleaseAllCallsAcceptWaitingCall (0x556)

All calls (the active and the held) are released. The waiting call is accepted. MPReleaseAllCallsExceptWaitingCall can not be used if there is no waiting call.

2.1.80.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	MPReleaseAllCallsAcceptWaitingCall (0x556)	Result	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.80.2 Parameter

2.1.81 MPExplicitCallTransfer (0x557)

Connect (join) the held and active call, and disconnect the subscriber from both calls. This function cannot be used if the held or active call is a conference. MPExplicitCallTransfer can not be used if there is neither an active nor a held call.

2.1.81.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	MPExplicitCallTransfer (0x557)	Result	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.81.2 Parameter

2.1.82 CCConferenceCall (0x558)

The property CCConferenceCall contains all Call-IDs of calls belonging to a conference. Therefore this function can be used to determine all involved calls in a conference. A conference in GSM may contain up to 6 members in addition to myself.

2.1.82.1 Format of Function

Function classes: Array of { Boolean }

FBlock	Function	OPType	Parameter
Telephone (0x50)	CCConferenceCall (0x558)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.82.2 Parameter

Pos

pos is equal to call ID Valid range: x=0..NMAX, y=0

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

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	MemberOfConference[1], MemberOfConference[2], ..., MemberOfConference[Nmax]
		{ x>0 }	MemberOfConference[x]

MemberOfConference

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	no member of conference
		True	member of conference
	Bit 1 ... 7	-	reserved

2.1.83 CCSplit (0x559)

Place all active calls of a conference on hold except the call that shall remain active. This function can only be used, if there is an active conference and no call on hold or if there is a conference on hold and no call active. The call that shall remain active is referenced by its Call-ID.

2.1.83.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	CCSplit (0x559)	Result	
		StartResult	CallID
		Error	ErrorCode, ErrorInfo

2.1.83.2 Parameter

CallID

Valid range: 1..NMAX

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

2.1.84 CCJoin (0x55A)

Add a held call to an active call (Join). Either the held or the active call may be a conference. The new conference is active afterwards.

2.1.84.1 Format of Function

Function classes: Trigger

FBlock	Function	OPType	Parameter
Telephone (0x50)	CCJoin (0x55A)	Result	
		StartResult	
		Error	ErrorCode, ErrorInfo

2.1.84.2 Parameter

2.1.85 WaitingCallAlert (0x560)

Switch on/off waiting call signalling

2.1.85.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	WaitingCallAlert (0x560)	Processing	
		Result	
		StartResult	OnOff
		Error	ErrorCode, ErrorInfo

2.1.85.2 Parameter

OnOff

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

2.1.86 OpenDataConnection (0x600)

This function builds a data connection to the given phone number.

2.1.86.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	OpenDataConnection (0x600)	Processing	
		Result	CallID
		Start	Speed, Type, ECC, Compression, Data Mode, TelNumber, Transparent
		StartResult	Speed, Type, ECC, Compression, Data Mode, TelNumber, Transparent
		Error	ErrorCode, ErrorInfo

2.1.86.2 Parameter

Transparent

Connection Type

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Non Transparent Connection
		True	Transparent Connection
	Bit 1 ... 7	-	reserved

Data Mode

Data mode for the connection to be established

Basis datatype	Range of values	Code	Description
Enum	0x00..0x03	0x00	synchron
		0x01	asynchron
		0x02	PAD Access
		0x03	Packet Access

Speed

0000 = 300 bps 0001 = 600 bps 0010 = 1200 bps 0011 = 2400 bps 0100 = 4800 bps 0101 = 9600 bps 0110 = 14400 bps 0111 = 19200 bps 1000 = 28800 bps 1001 = 38400 bps 1010 = 64000 bps 1011 = 128000 bps 1111 = Auto select (fastest possible) other values = direct speed in bps

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

CallID

Valid range: 1..NMAX

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

TelNumber

The number to dial

Basis datatype	MaxSize
String	40

Compression

Data compression

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Data Compression Off
		True	Data Compression On
	Bit 1 ... 7	-	reserved

ECC

Error Correction

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Error Correction Off
		True	Error Correction On
	Bit 1 ... 7	-	reserved

Type

Type of connection to be established

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	data connection V34
		0x01	data connection V 110
		0x02	Fax Class 1
		0x03	Fax Class 2
		0x04	Fax Class 2.0

2.1.87 EndDataConnection (0x601)

Finish one or all existing data connections

2.1.87.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	EndDataConnection (0x601)	Processing	
		Result	
		Start	CallID
		StartResult	CallID
		Error	ErrorCode, ErrorInfo

2.1.87.2 Parameter

CallID

Valid Range: 1..NMAX CallID=0xFF ends all connections

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

2.1.88 AcceptDataConnection (0x602)

Answers an incoming data call

2.1.88.1 Format of Function

Function classes: Unclassified Method

FBlock	Function	OPType	Parameter
Telephone (0x50)	AcceptDataConnection (0x602)	Processing	
		Result	
		Start	CallID
		StartResult	CallID
		Error	ErrorCode, ErrorInfo

2.1.88.2 Parameter

CallID

Valid Range: 1..NMAX

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

2.1.89 DataCapabilities (0x603)

With the property DataCapabilities the possible modes for data transmission the telephone supports can be inquired.

2.1.89.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
Telephone (0x50)	DataCapabilities (0x603)	Get	
		Status	Types, Speeds, ECC, Compression, Data Mode, ConnectionType
		Error	ErrorCode, ErrorInfo

2.1.89.2 Parameter

ConnectionType

Connection Type

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Non Transparent Connection
		True	Transparent Connection
	Bit 1 ... 7	-	reserved

Data Mode

Data mode for the connection to be established

Basis datatype	Range of values	Code	Description
Enum	0x00..0x03	0x00	synchron
		0x01	asynchron
		0x02	PAD Access
		0x03	Packet Access

Speeds

Supported line speeds 0000 = 300 bps 0001 = 600 bps 0010 = 1200 bps 0011 = 2400 bps
0100 = 4800 bps 0101 = 9600 bps 0110 = 14400 bps 0111 = 19200 bps 1000 = 28800 bps
1001 = 38400 bps 1010 = 64000 bps 1011 = 128000 bps 1111 = Auto select (fastest possible)
device supports nonstandard Bitrates

Basis datatype	Range of values	Code	Description
Enum	0x00..0x0F	0x00	300 bps
		0x01	600 bps
		0x02	1200 bps
		0x03	2400 bps
		0x04	4800 bps
		0x05	9600 bps
		0x06	14400 bps
		0x07	19200 bps
		0x08	28800 bps
		0x09	38400 bps
		0x0A	64000 bps
		0x0B	128000 bps
		0x0F	Auto select (fastest possible) device supports nonstandard Bitrates

Compression

Data compression

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Data Compression Off
		True	Data Compression On
	Bit 1 ... 7	-	reserved

ECC

Error Correction

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Error Correction Off
		True	Error Correction On
	Bit 1 ... 7	-	reserved

Types

Available connection types

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	data connection V34
		0x01	data connection V 110
		0x02	Fax Class 1
		0x03	Fax Class 2
		0x04	Fax Class 2.0

2.1.90 ConnectionStatus (0x604)

With the property ConnectionStatus the current status of the data connection can be inquired.

2.1.90.1 Format of Function

Function classes: Unclassified Property

FBlock	Function	OPType	Parameter
Telephone (0x50)	ConnectionStatus (0x604)	Get	CallID
		Status	CallID, Type, ECC, Speed, Compression, Data Mode, Transparent
		Error	ErrorCode, ErrorInfo

2.1.90.2 Parameter

Transparent

Connection Type

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Non Transparent Connection
		True	Transparent Connection
	Bit 1 ... 7	-	reserved

Data Mode

Data mode for the connection to be established

Basis datatype	Range of values	Code	Description
Enum	0x00..0x03	0x00	synchron
		0x01	asynchron
		0x02	PAD Access
		0x03	Packet Access

Speed

0000 = 300 bps 0001 = 600 bps 0010 = 1200 bps 0011 = 2400 bps 0100 = 4800 bps 0101 = 9600 bps 0110 = 14400 bps 0111 = 19200 bps 1000 = 28800 bps 1001 = 38400 bps 1010 = 64000 bps 1011 = 128000 bps 1111 = Auto select (fastest possible) other values = direct speed in bps

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

CallID

Valid range: 1..NMAX ID of the connection for which the status should be returned

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

Compression

Data compression

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Data Compression Off
		True	Data Compression On
	Bit 1 ... 7	-	reserved

ECC

Error Correction

Basis datatype	Bit #	Code	Description
Boolean	Bit 0	False	Error Correction Off
		True	Error Correction On
	Bit 1 ... 7	-	reserved

Type

Type of connection to be established

Basis datatype	Range of values	Code	Description
Enum	0x00..0x04	0x00	data connection V34
		0x01	data connection V 110
		0x02	Fax Class 1
		0x03	Fax Class 2
		0x04	Fax Class 2.0

2.1.91 DataSpeedCapabilities (0x605)

This property administrates the list of available data speeds

2.1.91.1 Format of Function

Function classes: Array of { Number }

FBlock	Function	OPType	Parameter
Telephone (0x50)	DataSpeedCapabilities (0x605)	Get	Pos
		Status	Pos, Data
		Error	ErrorCode, ErrorInfo

2.1.91.2 Parameter

Pos

The parameter Pos consists of two byte and shows which parameter hasll be set, requested or read. Valid range: x=0..NMAX, y=0

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

Data

Basis datatype	Length	Description	
Stream	-	Pos	Data
		{ x=0 }	Speed[1], ..., Speed[NMax]
		{ x>0 }	Speed[x]

Speed

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

3 FunctionBlock Dynamic Specification

TBD

