MTi and MTx
Low-Level Communication Documentation

Document MT0101P
Revision A, June 3rd 2005

Xsens Technologies B.V.
Capitool 50
P.O. Box 545
7500 AM Enschede
The Netherlands

phone +31-(0)53-4836444
fax +31-(0)53-4836445
e-mail support@xsens.com
internet www.xsens.com

Revisions

Revision	Date	By	Changes
A	June 3rd 2005	SS	First version

Table of Contents

1 Introduction

This document describes how to communicate with the MTi / MTx directly on low-level basis. The communication which is based on messages enables the user to change the configuration of the MTi / MTx and retrieve the output data.

The configuration settings are all user-settable. Examples are sample frequency, in- & output synchronization, baudrate and output modes. The different output modes enable the user to change the output data to the one that is preferred.

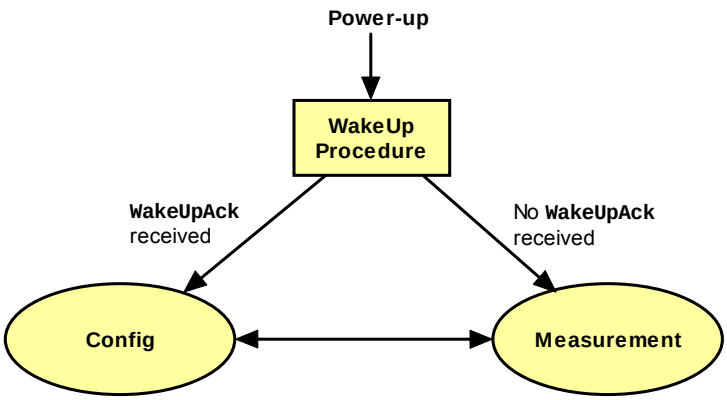
Configuration changes are executed in the so-called Config State. In this state the MTi / MTx accepts messages that set the output mode or other settings. Whenever the preferred configuration is set the user can set the MTi / MTx into the Measurement State. In this state the MTi / MTx start output the data based the current configuration settings. The MTi / MTx states are discussed in the chapter 2.

The messages used in Config / Measurement state are described in chapter 3. First the generic format of a message is explained. After this the messages all described per State.
Chapter 3.3 gives some examples of how to change the configuration and shows some of the possibilities of the MTi / MTx.The last chapter gives a quick overview of the MTi / MTx messages with a short description, see chapter 4.

2 States

The MTi / MTx has two states, i.e. Config and Measurement State. In the Config State various settings can be read & written and in the Measurement State the MTi / MTx will output its data message which contains data dependent on the current configuration.

There are two different ways to enter the Config State or the Measurement State. At power-up the MTi / MTx starts the WakeUp procedure and it will send the **`WakeUp`** message. If no action is taken the device enters the Measurement State. But if the **`WakeUpAck`** message is sent within 500ms after reception of the **`WakeUp`** message the MTi / MTx enters the Config State. Just before entering the Measurement State, the **`Configuration`** message is sent to the host. This is the configuration that is read from the internal memory and will be used in the Measurement State. Another way to enter the Config or Measurement State is to use the **`GoToConfig`** or **`GoToMeasurement`** messages while the other state is active.



2.1 Config State

Config State is used to get and/or set various settings of the MTi / MTx. Most of the settings will change the configuration which defines the device functionality in Measurement State. Settings that change the configuration are for example the sample period, output mode or synchronization properties.

At power-up all settings are read from non-volatile memory. Settings changed in Config State are immediately stored in the memory and will retain their latest values even if the device is disconnected from power. Some messages have an additional parameter that specifies whether or not the new values should be stored in non-volatile memory. Either way, the setting changes are direct. There is one exception, namely the baudrate setting. The new setting will be used at the next power-cycle or after a reset.

See chapter 3 for more information about messages.

2.2 Measurement State

In Measurement State the MTi / MTx will output its data to the host in a way depending on the configuration settings defined in Config State. A single message, **`MTData`**, is used for all different output modes. It is therefore important that the host knows how the device is configured. The configuration depends for the data must be interpreted. A special message, **`Configuration`**, contains the relevant information which with the data received by the host in Measurement State can be unambiguous interpreted. When logging **`MTData`** messages it is advisable to include the **`Configuration`** message in the data for future analysis.

If the host does not respond to the **`WakeUp`** message at power-up (or after a reset) the MTi / MTx will automatically enter the Measurement State. Just before entering the state it will send the **`Configuration`** message. The configuration settings are all read from the non-volatile memory and are used during the measurement. The default configuration of the MTi / MTx is shown in the following table.

Property	Value
Output mode	Orientation output
Output settings	Orientation in quaternion mode Sample counter
Sample frequency	100 Hz
Baudrate	115k2 bps
Output skip factor	0
SyncIn	Disabled
SyncOut	Disabled

Measurement State is normally not used to change any settings. However, there is one exception. The **`SetFilterParameters`** message can also be used during the measurement to dynamically adjust the filter parameters used to calculate the orientation. To change other settings the device must enter the Config State and the user must first send the **`GoToConfig`** message.

See chapter 3 for more information about messages.

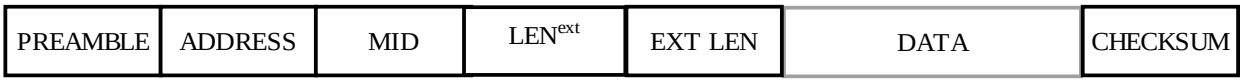
3 Messages

3.1 Message structure

The communication with the MTi and MTx is done by messages which are built according to a standard structure. The message has two structures; one with a standard length and one with extended length. The standard length message has a maximum of 254 data bytes and is used most frequently. In some cases the extended length message needs to be used if the number of data bytes exceeds 254 bytes.

An Xbus message (standard length) contains the following fields:

An Xbus message (extended length) contains these fields:



Field	Field width	Description
Preamble	1 byte	Indicator of start of packet → 250 (0xFA)
Address	1 byte	Address value → 255 (0xFF)
MID	1 byte	Message identifier
LEN	1 byte	For standard length message: Value equals number of bytes in DATA field. Maximum value is 254 (0xFE) For extended length message: Field value is always 255 (0xFF)
EXT LEN	2 bytes	16 bit value representing the number of data bytes for extended length messages. Maximum value is 2048 (0x0800)
DATA (standard length)	0 – 254 bytes	Data bytes (optional)
DATA (extended length)	255 – 2048 bytes	Data bytes
Checksum	1 byte	Checksum of message

Preamble

Every message starts with the preamble. This field always contains the value 250 (=0xFA).

Address

All messages used for the MTi and MTx use the address value 255 (0xFF) indicating a “master device”.

Message Identifier (MID)

This message field identifies the kind of message. For a complete listing of all possible messages see chapter 3.3.

Length (LEN)
Specifies the number of data bytes in the DATA field for standard length message. If value 255 (=0xFF) is specified the message will be interpreted as an extended message length and the next two bytes are used for the number of bytes in the DATA field. If zero, no DATA field exists.

Extended Length (EXT LEN)
This field is a 16 bit value representing the number of data bytes in the DATA field of an extended length message.

Data (DATA)
This field contains the data bytes and it has a variable length which is specified in the Length or Extended Length field. The interpretation of the data bytes is message specific, i.e. depending on the MID value the meaning of the data bytes is different. See the description of the specific message for more details about the data bytes.

Checksum
This field is used for communication error-detection. If all message bytes excluding the preamble are summed and the lower byte value of the result equals zero, the message is valid and it may be processed. The checksum value of the message should be included in the summation.

3.2 Message usage
Generally, a message with a certain MID value will be replied with a message with a MID value that is increased by one, i.e. the acknowledge message. Depending on the message type the acknowledge message can have a data field (no fixed length) or not. If nothing is specified the data field does not exist. In some cases an error message will be returned (MID = 66 (0x42)). This occurs in case the previous message has invalid parameters, is not valid, or could not be successfully executed. An error message contains an error code in its data field.

Example
Requesting the device ID of a MTi / MTx:

Sending message:
`ReqDID =` 0xFA 0xFF 0x00 0x00 0x01 (hexadecimal values)
Receiving message (= Acknowledge):
`DeviceID =` 0xFA 0xFF 0x01 0x04 HH HL LH LL CS (hexadecimal values)

The requested Device ID is given in the acknowledge message **`DeviceID`** (here shown as: HH HL LH LL, the checksum is CS). As you can see the MID (Message ID) of the acknowledge message is increased by one in comparison with the sending message **`ReqDID`**.

Some messages have the same MID and depending on whether or not the message contains the data field the meaning differs. This is the case with all the messages that refer to changeable settings. For example, the MID of message requesting the output mode (**`ReqOutputMode`**) is the same as the message that sets the output mode (**`SetOutputMode`**). The difference between the two messages is that the Length field of **`ReqOutputMode`** is zero and non-zero for **`SetOutputMode`**.

Example
Request current output mode:

Sending message:
`ReqOutputMode =` 0xFA 0xFF 0xD0 0x00 0x31 (hexadecimal values)
Receiving message (= Acknowledge):
`ReqOutputModeAck =` 0xFA 0xFF 0xD1 0x02 MH ML CS (hexadecimal values)

`ReqOutputModeAck` contains data which represents the current mode (= MH & ML). CS stands for the checksum value. To change the output mode you must add the new mode in the data field of the sending message:

Set the output mode:

Sending message:
`SetOutputMode =` 0xFA 0xFF 0xD0 0x02 MH ML CS (hexadecimal values)
Receiving message (= Acknowledge):
`SetOutputModeAck =` 0xFA 0xFF 0xD1 0x00 0x30 (hexadecimal values)

3.3 Message listing

3.3.1 WakeUp + State messages

`WakeUp`

MID	62 (0x3E)
DATA	n/a
Direction	To host
Valid in	WakeUp procedure

At power-up or after issuing a reset this message is sent to the host. If the host sends **`WakeUpAck`** (MID 63 (0x3F)) within 500ms after reception of this message, the MTi / MTx enters the Config State else Measurement State.

`GoToConfig`

MID	48 (0x30)
DATA	n/a
Direction	To MTi / MTx
Valid in	Measurement State & Config State

Switch the active state of the device from Measurement State to Config State. This message can also be used in Config State to confirm that Config State is currently the active state.

`GoToMeasurement`

MID	16 (0x10)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Switch the active state of the device from Config State to Measurement State. The current configuration settings are used to start the measurement.

Reset

MID	64 (0x40)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State & Measurement State

Sending this message will cause the MTi / MTx to reset and to activate the WakeUp procedure. A acknowledge message will be send to confirm reception of the **Reset** message.

3.3.2 Informational messages

ReqDID

MID	0 (0x00)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request to send the device identifier (or serial number). MTi / MTx acknowledges by sending the **DeviceID** message.

DeviceID

MID	1 (0x01)
DATA	IDHH IDHL IDLH IDLL (4 bytes)
Direction	To host
Valid in	Config State

Acknowledge of **ReqDID** message. Data field contains device ID / serial number.

InitBus

MID	2 (0x02)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

This message is supported so the host can use the same message for MTi / MTx as for the XbusMaster. The device will answer with the **InitBusResults** message. For the MTi / MTx it has the same functionality as **ReqDID**.

InitBusResults

MID	3 (0x03)
DATA	IDHH IDHL IDLH IDLL (4 bytes)
Direction	To host
Valid in	Config State

Acknowledge of **InitBus** message. Data field contains device ID / serial number and corresponds to the S/N number on the bottom of the device (hexadecimal values).

ReqFWRev

MID	18 (0x12)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request to send the firmware revision of the device. MTi / MTx acknowledges by sending **FirmwareRev** message.

FirmwareRev

MID	19 (0x13)
DATA	MAJOR MINOR REV (3 bytes)
Direction	To host
Valid in	Config State

Acknowledge of **ReqFWRev** message. Data field contains firmware code (major, minor, revision part).

ReqDataLen

MID	10 (0x0a)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request the length of the data field of the **MTData** message based on the current configuration. The device acknowledges with the **DataLen** message.

DataLen

MID	11 (0x0b)
DATA	DATALENGTH (2 bytes)
Direction	To host
Valid in	Config State

Acknowledge of **ReqDataLen** message. The data field contains DATALENGTH, an unsigned 16 bits value that is equal to the data length of the **MTData** message send in Measurement State based on the current configuration.

Error

MID	66 (0x42)
DATA	ERRORCODE (1 byte)
Direction	To host
Valid in	Config & Measurement State

Indicate that an error has occurred. Error type is specified in the ERROR field.

ERRORCODE
A one-byte value indicating the type of error. See table.

ERRORCODE	Error description
3 (0x03)	Period sent is not within valid range
4 (0x04)	Message sent is invalid
30 (0x1E)	Timer overflow, this can be caused to high sample frequency or sending to much data to MTi / MTx during measurement
32 (0x20)	Baudrate sent is not within valid range
33 (0x21)	Parameter sent is invalid or not within range

3.3.3 Device-specific messages

ReqBaudrate

MID	24 (0x18)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request the baudrate of the device. See **SetBaudrate** for data field description of the received acknowledge.

SetBaudrate

MID	24 (0x18)
DATA	BAUDRATE (1 byte)
Direction	To MTi / MTx
Valid in	Config State

This message changes the baudrate of the communication interface (RS-232 or RS-422). The new baudrate will be stored in non-volatile memory and will become active after a reset or power cycle.

BAUDRATE
See table for the different baudrates and the corresponding value of BAUDRATE. The default baudrate is 115k2 bps.

Baudrate (bps)	BAUDRATE
921k6	128 (0x80)
460k8	0 (0x0)
230k4	1 (0x1)
115k2	2 (0x2)
76k8	3 (0x3)
57k6	4 (0x4)
38k4	5 (0x5)
28k8	6 (0x6)
19k2	7 (0x7)
14k4	8 (0x8)
9k6	9 (0x9)

ReqErrorMode

MID	218 (0xDA)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request the current error mode - see **SetErrorMode** for information about data field of received acknowledge.

SetErrorMode

MID	218 (0xDA)
DATA	ERRORMODE (2 bytes)
Direction	To MTi / MTx
Valid in	Config State

Set the error mode to a specific ERRORMODE.

ERRORMODE
The ERRORMODE is an unsigned 16 bit value that defines how the device should deal with errors. The default error mode is that in case the sampling instance is missed the sample counter is increase and no further action is taken (ERRORMODE = 1).

ERRORMODE	Description
0 (0x0000)	Ignore any errors
1 (0x0001)	Error message is not sent In case of missing sampling instance: increase sample counter
2 (0x0002)	Error message is sent In case of missing sampling instance: increase sample counter
3 (0x0003)	Error message is sent and the device will enter the Config State (if in Measurement State)

ReqLocationID

MID	132 (0x84)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request location ID - see **SetLocationID** for information about data field of received acknowledge message.

SetLocationID

MID	132 (0x84)
DATA	LOCID (2 bytes)
Direction	To MTi / MTx
Valid in	Config State

Set a user-defined value. This value can be used to give the device a location dependant identifier.

LOCID
A 16 bit value having an arbitrary value set by the user. Default value is zero.

`RestoreFactoryDef`

MID	14 (0x0e)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

If this message is sent to the MTi / MTx the factory defaults are restored. All settings that have changed will be discarded including object alignments, filter settings, etc.

3.3.4 Synchronization messages

`ReqSyncInSettings`

MID	214 (0xD6)
DATA	PARAM (1 byte)
Direction	To MTi / MTx
Valid in	Config State

Request the one of the current SyncIn settings of the device, i.e. mode, skip factor or offset. The requested setting is specified by the PARAM value. See **`SetSyncInSettings`** for description of the data field of the received acknowledge.

PARAM
PARAM defines which of the SyncIn settings should be returned by the acknowledge message. See following table.

PARAM value	Setting
0 (0x00)	SyncIn mode
1 (0x01)	Skip factor
2 (0x02)	Offset

`SetSyncInSettings`

MID	214 (0xD6)
DATA	PARAM VALUE (1+x bytes)
Direction	To MTi / MTx
Valid in	Config State

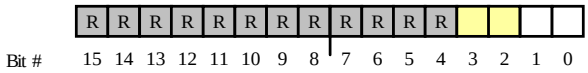
Set a SyncIn setting defined by the PARAM field with the value specified with the VALUE field. Use SyncIn to synchronize the MTi / MTx with external hardware. Both the MTi (standard version) and MTx have an input pin that can be connected to other hardware, like a high-precision clock generator. See **MTi and MTx User Manual (and Technical Documentation)** for more information about the SyncIn pin.

PARAM

The PARAM value (1 byte) indicates which SyncIn setting will be set. The VALUE field, which can be either two or four bytes long, defines the new value of the setting. See table for the different PARAM values.

PARAM value	Setting
0 (0x00)	SyncIn mode
1 (0x01)	Skip factor
2 (0x02)	Offset

VALUE - SyncIn mode



PARAM	VALUE	Description
0		SyncIn mode (2 bytes)
	Bit 1-0	Trigger mode
		00 = SyncIn mode disabled (default)
		01 = Rising Edge
		10 = Falling Edge
		11 = Reserved
	Bit 3-2	Trigger type
		00 = ADC sampling (incl sending of **MTData**)
		01 = Sending of latest output data
	Bit 15-4	Reserved

VALUE – Skip factor

An unsigned 16 bit value that defines how many times the trigger should be skipped before actually trigger the device.

PARAM	VALUE	Description
1		Skip factor (2 bytes)
	Bit 15-0	Skip factor value

VALUE – Offset

An unsigned 32 bit value that defines the number of clock ticks to delay the triggering after a valid detection. One clock tick is equal to 1 / 29.4912 MHz = 33.9ns.

PARAM	VALUE	Description
2		Offset (4 bytes)
	Bit 31-0	Offset value

ReqSyncOutSettings

MID	216 (0xD8)
DATA	PARAM (1 byte)
Direction	To MTi
Valid in	Config State

Request the one of the current SyncOut settings of the MTi, i.e. SyncOut mode, skip factor, offset or pulse width. The requested setting is specified by the PARAM value. See **SetSyncOutSettings** for description of the data field of the received acknowledge.

PARAM

PARAM defines which of the SyncOut settings should be returned by the acknowledge message. See following table.

PARAM value	Setting
0 (0x00)	SyncOut mode
1 (0x01)	Skip factor
2 (0x02)	Offset
3 (0x03)	Pulse width

SetSyncOutSettings

MID	216 (0xD8)
DATA	PARAM VALUE (1+x bytes)
Direction	To MTi
Valid in	Config State

Set a SyncOut setting defined by the PARAM field with the value specified with the VALUE field. Use SyncOut to synchronize other external hardware with the MTi. Only the standard version of the MTi has an output pin that can be connected to other hardware which can be synchronized with the MTi. See **MTi and MTx User Manual (and Technical Documentation)** for more information about the SyncOut pin.

PARAM

The PARAM value (1 byte) indicates which SyncOut setting will be set. The VALUE field, which can be either two or four bytes long, defines the new value of the setting. See table for the different PARAM values.

PARAM value	Setting
0 (0x00)	SyncOut mode
1 (0x01)	Skip factor
2 (0x02)	Offset
3 (0x03)	Pulse width

VALUE - SyncOut mode

See the next table for the description of the SyncOut mode bits.

PARAM	VALUE	Description
0		SyncOut mode (2 bytes)
	Bit 3-0	
		0000 = SyncOut mode disabled (default)
		0001 = Toggle mode
		0010 = Pulse mode
	Bit 4	Polarity
		0 = Negative pulse
		1 = Positive pulse
	Bit 15-5	Reserved

VALUE – Skip factor
An unsigned 16 bit value that defines how many times the internal sampling instance should be skipped before changing the SyncOut pin state. The internal trigger instance is based on the sampling period set by **`SetPeriod`**.

PARAM	VALUE	Description
1		Skip factor (2 bytes)
	Bit 15-0	Skip factor value

VALUE – Offset
An unsigned 32 bit value that defines the number of clock ticks to delay the SyncOut state-change after the internal sampling instance. One clock tick is equal to 1 / 29.4912 MHz = 33.9ns.

PARAM	VALUE	Description
2		Offset (4 bytes)
	Bit 31-0	Offset value
		Valid range is 0 & 513 (0x00000201) and higher

VALUE – Pulse width
An unsigned 32 bit value that defines the pulse width when SyncOut mode is set to Pulse mode. The pulse width is specified in ticks and one tick is equal to 1 / 29.4912 MHz = 33.9ns.

PARAM	VALUE	Description
3		Pulse width (4 bytes)
	Bit 31-0	Pulse width value
		Valid range is 0 & 1700 (0x000006A4) and higher

3.3.5 Configuration messages

`ReqConfiguration`

MID	12 (0x0c)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request the configuration settings of the device. Can be used for logging purposes - include the **`Configuration`** message in the log file to store the settings for offline processing of the data.

Configuration

MID	13 (0x0d)
DATA	CONFIGURATION (118 bytes)
Direction	To host
Valid in	Config State

Acknowledge of **ReqConfiguration** message. Data field contains the current configuration of the MTi / MTx.

CONFIGURATION
The CONFIGURATION data contains the following information. For more information about the different fields check the corresponding message description.

offset (B)	length (B)	field description
0	4	Master device ID
4	2	Sampling period
6	2	Output skip factor
8	2	Syncin settings - Mode
10	2	Syncin settings - Skip Factor
12	4	Syncin settings - Offset
16	8	String with format YYYYMMDD (can be set by host)
24	8	Time with format HHMMSSHH (can be set by host)
32	32	Reserved (host)
64	32	Reserved (client)
96	2	Num of sensors (= 1 (0x0001))
98	4	Device ID (same as master device ID)
102	2	Data length of MTData message
104	2	Output mode
106	4	Output settings
110	8	Reserved

ReqPeriod

MID	4 (0x04)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request the current sample period. The MTi / MTx replies with **ReqPeriodAck**. The data field of this message contains the sample period. For the description of the data field see **SetPeriod**.

SetPeriod

MID	4 (0x04)
DATA	PERIOD (2 bytes)
Direction	To host
Valid in	Config State

Sets the sampling period of the device used in Measurement State.

PERIOD
PERIOD is an unsigned 16-bit value indicating the length of the period. Resolution is in (1/115200) seconds, i.e. 8.68 us. The following table shows the default, minimum and maximum values.

PERIOD	Value	Sampling period (freq)
Default	1152 (0x0480)	10.0ms (100Hz)
Minimum	225 (0x00E1)	1.95ms (512Hz)
Maximum	11520 (0x2D00)	100ms (10Hz)

The MTi / MTx outputs the **MTData** at a rate that is not only depending on the sampling frequency but also on the OutputSkipfactor (see **SetOutputSkipfactor**). Normally this factor is zero and the **MTData** message is sent (1 / sampling period) times per second. A value higher than zero corresponds to how many times the **MTData** message is NOT sent to the host. To calculate how often the **MTData** is sent to the host, use the following formula.

MTData frequency (Hz) = 115200 / (PERIOD * (OutputSkipfactor + 1))

For the MTi / MTx sample frequencies lower than 100Hz are not setable directly. The device standard uses 100Hz as lowest sampling frequency. However in combination with the OutputSkipfactor (see **SetOutputSkipfactor** message) lower frequencies can be set. For example, if **SetPeriod** is sent with a sampling period of 20ms (50Hz), the device will automatically set the sampling period to 10ms (100Hz) and the OutputSkipfactor to 1. The resulting sampling period is 10ms * (OutputSkipfactor + 1) = 20ms (50Hz). If the sampling period can not be made (OutputSkipfactor is not an integer), **SetPeriod** will return an error message. In this case, choose a lower sampling period with an integer OutputSkipfactor to generate the requested frequency. For example, to have a resulting sampling period of 13.33ms (75Hz) set the sampling period to 6.67ms (150Hz) and the OutputSkipfactor to 1.

ReqOutputSkipFactor

MID	212 (0xD4)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request how many times the data output is skipped before sending the data in the **MTData** message to host. For information about data field of received acknowledge see **SetOutputSkipFactor**.

SetOutputSkipfactor

MID	212 (0xD4)
DATA	SKIPFACTOR (2 bytes)
Direction	To MTi / MTx
Valid in	Config State

Set the output skip factor.

SKIPFACTOR
The skip factor is an unsigned 16 bit value and is by default zero. The value represents how many times the data output is skipped (running at the current sampling frequency) before the next **MTData** message is sent. I.e. at sample period of 5.0ms (200Hz) and a skip factor of 4, the measurement is running at 200Hz but the data is send at a rate of 40Hz (NOT 50Hz!!). See also **SetPeriod** for more information about the relationship between the sampling period and output skip factor.

If SKIPFACTOR is set to 65535 (0xFFFF), no data will be sent to the host and **ReqData** can be use to request an **MTData** message at an arbitrary time.

SKIPFACTOR	Description
≠ 65535 (0xFFFF)	Skipfactor value
65535 (0xFFFF)	Do not send **MTData** automatically

ReqOutputMode

MID	208 (0xD0)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request the current output mode. See **SetOutputMode** for information about data field of the received acknowledge.

SetOutputMode

MID	208 (0xD0)
DATA	MODE (2 bytes)
Direction	To MTi / MTx
Valid in	Config State

Sets the output mode of the MTi / MTx. The output mode can be set to calibrated sensor data and/or orientation data. The un-calibrated raw data output however can not be used together with any of the other outputs.

MODE

MODE bits	Output mode
Bit 1	Calibrated data
Bit 2	Orientation data
Bit 14	Un-calibrated raw data (not in combination with calibrated sensor data and/or orientation data)

ReqOutputSettings

MID	210 (0xD2)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

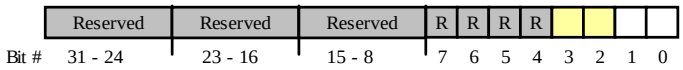
Requests the current output settings. See **SetOutputSettings** for information about data field of received acknowledge.

SetOutputSetings

MID	210 (0xD2)
DATA	SETTINGS (4 bytes)
Direction	To MTi / MTx
Valid in	Config State

Sets the output settings of the MTi / MTx..

SETTINGS



SETTINGS bits	Settings
Bit 1-0	Timestamp output
	00 = No timestamp
	01 = Sample Counter
Bit 3-2	Orientation Mode
	00 = Quaternion
	01 = Euler angles
	10 = Matrix

3.3.6 Data-related messages

ReqData

MID	52 (0x34)
DATA	n/a
Direction	To MTi / MTx
Valid in	Measurement State

This message can be used to ask the MTi / MTx to send data to the host. Normally, the MTi / MTx will send the **MTData** message automatically according to the sampling period & output skip factor settings. If however, the OutputSkipFactor is set to 65535 (0xFFFF) the MTi / MTx will not send the **MTData** except if the user sends **ReqData**.

MTData

MID	50 (0x32)
DATA	DATA (length variable)
Direction	To host
Valid in	Measurement State

This message contains the output data depending on the current Output Mode & Output settings. The data field can contain multiple data outputs but the order of outputs is always the same. The following order is used (disabled outputs must be omitted):

1. Un-calibrated raw data output
2. Calibrated data output
3. Orientation data output
4. Sample counter

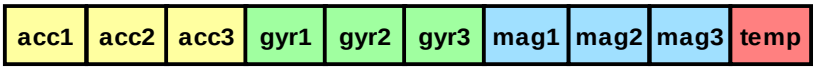
Un-calibrated raw data output can not be used together with calibrated and/or orientation data output. The following text explains the data values of each output.

DATA

The data can contain multiple outputs. All the different outputs are described separately here. If not specified otherwise each data value is 4 bytes long and corresponds with the single-precision floating-point value as defined in the IEEE 754 standard (= float).

Un-calibrated raw data output mode (20 bytes)

Contains the un-calibrated raw data output of the accelerations, rate of turn and magnetic field in X, Y & Z axes. These values are equal to the analog-digital converter readings of the internal sensors. The data values are NOT float values but 16 bit unsigned integer values.

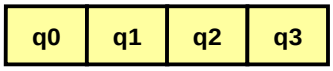


Calibrated data output mode (36 bytes)

Contains the calibrated data output of the accelerations, rate of turn and magnetic field in X, Y & Z axes in floats.

Orientation data output mode – quaternion (16 bytes)

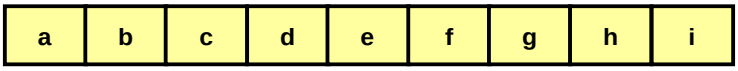
Contains the q0, q1, q2 and q3 quaternions, in floats, that represent the orientation of the MTi / MTx



Orientation data output mode – Euler angles (12 bytes)

Contains the three Euler angles, in floats, that represent the orientation of the MTi / MTx

Orientation data output mode – Matrix (36 bytes)
Contains the rotation matrix (DCM), in floats, that represents the orientation of the MTi / MTx. See **MTi and MTx User Manual (and Technical Documentation)** for the interpretation of the data values.



Sample counter (2 bytes)
The (optional) sample counter is a 16 bit unsigned integer value that is increased for each transmission of the **MTData** message. If its maximum value is reached, i.e. 65535 (0xFFFF), it will wrap and start at zero again. See also **SetErrorMode** for the relation between the sample counter and the error mode.

3.3.7 Filter messages

ReqHeading

MID	130 (0x82)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Request the current heading setting - see **SetHeading** for information about data field of received acknowledge.

SetHeading

MID	130 (0x82)
DATA	HEADING (4 bytes)
Direction	To MTi / MTx
Valid in	Config State

Sets the heading with the value specified in data field.

HEADING
HEADING is a single-precision floating-point number (4 bytes).

ReqInitTrackMode

MID	136 (0x88)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Requests initial tracking mode - see **SetInitTrackMode** for information about data field of received acknowledge message.

SetInitTrackMode

MID	136 (0x88)
DATA	TRACKMODE (2 bytes)
Direction	To MTi / MTx
Valid in	Config State

Set the device to en- or disable the initial tracking mode.

TRACKMODE
A two byte value that defines whether or not the initial tracking mode should be enabled

TRACKMODE	Description
0 (0x0000)	Disable initial tracking mode
1 (0x0001)	Enable initial tracking mode

StoreFilterState

MID	138 (0x8A)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config State

Store the gyroscope bias & magnetometer bias in non-volatile memory. If InitTrackMode is disabled only the magnetometer bias is used.

ReqFilterParameters

MID	160 (0xA0)
DATA	PARAM (1 byte)
Direction	To MTi / MTx
Valid in	Config State & Measurement State

Request to send a filter parameter defined by PARAM which is used to calculate the orientation. For more information about data field format of this message and/or receiving acknowledge see **SetFilterParameters** (the store field is not included in the data field).

PARAM
PARAM defines which of the FilterParameters should be returned by the acknowledge message. See following table.

PARAM value	Setting
0 (0x00)	Gain
1 (0x01)	Magnetometer weighting factor

SetFilterParameters

MID	160 (0xA0)
DATA	PARAM FILTERDATA STORE (1+4+1 bytes)
Direction	To MTi / MTx
Valid in	Config State & Measurement State

Set the filter parameters. Use the STORE value to store the parameter into non-volatile memory or not. When requested, latest value is always returned. After a reset or power cycle, the filter parameter that has been saved with STORE set will be used.

PARAM
The PARAM value (1 byte) indicates which Filter parameter will be set. The FILTERDATA field defines the new value of the setting. See table for the different PARAM values.

PARAM value	Setting
0 (0x00)	Gain
1 (0x01)	Magnetometer weighting factor

FILTERDATA - Gain
The value is a single-precision floating-point number (4 bytes) which defines the filter gain.

PARAM	VALUE	Description
0		Gain (4 bytes)
	Bit 31-0	Gain value (float)
		Valid range is <0.01 ; 50.0]

FILTERDATA – Weighting factor
The value is a single-precision floating-point number (4 bytes) which defines the magnetometer weighting gain.

PARAM	VALUE	Description
1		Magnetometer weighting factor (4 bytes)
	Bit 31-0	Weighting factor value (float)
		Valid range is [0.0 ; 10.0>

STORE
This single byte specifies whether the filter parameter should also be stored in non-volatile memory. Either way, the new parameter value is directly used in the calculations. But after a power-cycle or a reset the parameters that are stored in non-volatile memory are used. You can not store the parameter in non-volatile memory while in Measurement State.

STORE	Description
0 (0x00)	Do not store the parameter in non-volatile memory
1 (0x01)	Store parameter in non-volatile memory (only in Config State)

ReqAMD

MID	162 (0xA2)
DATA	n/a
Direction	To MTi / MTx
Valid in	Config & Measurement State

Request whether or not the MTi / MTx should adjust for magnetic disturbances. For more information about data format of the received acknowledge message see **SetAMD**.

SetAMD

MID	162 (0xA2)
DATA	AMDSETTING (2 bytes)
Direction	To MTi / MTx
Valid in	Config & Measurement State

Set whether or not the MT is adjusting for magnetic disturbances

AMDSETTING
A two-byte value that en- or disables the AMD functionality. Default value: disable AMD. If the setting will be stored in non-volatile memory if the active state is Config State.

AMDSETTING	Description
0 (0x0000)	Disable AMD
1 (0x0001)	Enable AMD

ResetOrientation

MID	164 (0xA4)
DATA	RESET (2 bytes)
Direction	To MTi / MTx
Valid in	Config State & Measurement State

Reset the orientation. Different resets are supported see next tabel. For more information about the different resets see **MTi and MTx User Manual (and Technical Documentation)**. If and only if, this message is sent with the correct RESET value the current setting will be saved or else the next time the Measurement State becomes active the 'old' settings are used.

ORIENTATIONRESET
A two-byte value indicating which reset to perform during Measurement State. To store the present settings enter the Config State and send the same message again with RESET equal to zero.

ORIENTATIONRESET	Description
0 (0x0000)	Store current settings (only in config mode)
1 (0x0001)	Heading reset
2 (0x0002)	Global reset
3 (0x0003)	Object reset
4 (0x0004)	Align reset

4 Message Reference Listing

This chapter gives a quick reference of all the valid messages. For more information about the use of the messages see chapter 3.

WakeUp & State messages (Section 3.3.1)

Message	MID	Direction	Description
WakeUp	62 (0x3E)	To host	The device sends this message at power-up or reset
WakeUpAck	63 (0x3F)	To MTi/MTx	If received within 500ms after WakeUp device enters Config State else Measurement State
GoToConfig	48 (0x30)	To MTi/MTx	Device enters config state
GoToConfigAck	49 (0x31)	To host	Device acknowledges GoToConfig message
GoToMeasurement	16 (0x10)	To MTi/MTx	Device enters measurement state
GoToMeasurementAck	17 (0x11)	To host	Device acknowledges GoToMeasurement message
Reset	64 (0x40)	To MTi/MTx	Reset device
ResetAck	65 (0x41)	To host	Acknowledgement of Reset message

Informational messages (Section 3.3.2)

Message	MID	Direction	Description
ReqDID	0 (0x00)	To MTi/MTx	Host request device ID of the device
DeviceID	1 (0x01)	To host	Device acknowledges request by sending its ID
InitBus	2 (0x02)	To MTi/MTx	Request device ID of the device (for compatibility of Xbus Master users)
InitBusResults	3 (0x03)	To host	Same as DeviceID message
ReqFWRev	18 (0x12)	To MTi/MTx	Host requests firmware revision of device
FirmwareRev	19 (0x13)	To host	Device acknowledges request by sending its firmware revision
ReqDataLen	10 (0x0a)	To MTi/MTx	Request the number of data bytes in MTData message
DataLen	11 (0x0b)	To host	Contains the number of data bytes of the MTData message
Error	66 (0x42)	To host	Error message

Device-specific messages (Section 3.3.3)

Message	MID	Direction	Description
ReqBaudrate	24 (0x18)	To MTi/MTx	Requests current baudrate of the serial communication
ReqBaudrateAck	25 (0x19)	To host	Device returns baudrate of serial communication
SetBaudrate	24 (0x18)	To MTi/MTx	Host sets baudrate of serial communication
SetBaudrateAck	25 (0x19)	To host	Device acknowledges SetBaudrate message
ReqErrorMode	218 (0xDA)	To MTi/MTx	Request error mode
ReqErrorModeAck	219 (0xDB)	To host	Device returns error mode
SetErrorMode	218 (0xDA)	To MTi/MTx	Host sets error mode
SetErrorModeAck	219 (0xDB)	To host	Device acknowledges SetErrorMode message
ReqLocationID	132 (0x84)	To MTi/MTx	Request location ID

ReqLocationIDAck	133 (0x85)	To host	Device returns location ID
SetLocationID	132 (0x84)	To MTi/MTx	Host sets location ID
SetLocationIDAck	133 (0x85)	To host	Device acknowledges SetLocationID message
RestoreFactoryDef	14 (0x0e)	To MTi/MTx	Restore factory defaults
RestoreFactoryDefAck	15 (0x0f)	To host	Device acknowledges RestoreFactoryDef message

Synchronization messages (Section 3.3.4)

Message	MID	Direction	Description
ReqSyncInSettings	214 (0xD6)	To MTi/MTx	Request a SyncIn setting of the device, i.e. SyncIn mode, skip factor or offset
ReqSyncInSettingsAck	215 (0xD7)	To host	Device returns SyncIn setting
SetSyncInSettings	214 (0xD6)	To MTi/MTx	Host sets a SyncIn setting
SetSyncInSettingsAck	215 (0xD7)	To host	Device acknowledges SetSyncInSettings message
ReqSyncOutSettings	216 (0xD8)	To MTi	Request a SyncOut setting of the device, i.e. SyncOut mode, skip factor, offset or pulse width
ReqSyncOutSettingsAck	217 (0xD9)	To host	Device returns SyncOut setting
SetSyncOutSettings	216 (0xD8)	To MTi	Host sets a SyncOut setting
SetSyncOutSettingsAck	217 (0xD9)	To host	Device acknowledges SetSyncOutSettings message

Configuration messages (Section 3.3.5)

Message	MID	Direction	Description
ReqConfiguration	12 (0x0c)	To MTi/MTx	Request the configuration of device. For logging/quick setup purposes
Configuration	13 (0x0d)	To host	Contains the configuration of device
ReqPeriod	4 (0x04)	To MTi/MTx	Request current sampling period
ReqPeriodAck	5 (0x05)	To host	Device returns sampling period
SetPeriod	4 (0x04)	To MTi/MTx	Host sets sampling period (10-500Hz)
SetPeriodAck	5 (0x05)	To host	Device acknowledges SetPeriod message
ReqOutputSkipFactor	212 (0xD4)	To MTi/MTx	Request how many times the data output is skipped before sending the MTData message
ReqOutputSkipFactorAck	213 (0xD5)	To host	Device returns OutputSkipFactor
SetOutputSkipfactor	212 (0xD4)	To MTi/MTx	Host sets OutputSkipFactor
SetOutputSkipfactorAck	213 (0xD5)	To host	Device acknowledges SetOutputSkipfactor message
ReqOutputMode	208 (0xD0)	To MTi/MTx	Request current output mode
ReqOutputModeAck	209 (0xD1)	To host	Device returns output mode
SetOutputMode	208 (0xD0)	To MTi/MTx	Host sets output mode
SetOutputModeAck	209 (0xD1)	To host	Device acknowledges SetOutputMode message
ReqOutputSettings	210 (0xD2)	To MTi/MTx	Request current output settings
ReqOutputSettingsAck	211 (0xD3)	To host	Device returns output settings
SetOutputSettings	210 (0xD2)	To MTi/MTx	Host sets output settings
SetOutputSettingsAck	211 (0xD3)	To host	Device acknowledges SetOutputSettings message

Data-related messages (Section 3.3.6)

Message	MID	Direction	Description
ReqData	52 (0x34)	To MTi/MTx	Host requests device to send MTData message
MTData	50 (0x32)	To host	Message with un-calibrated raw data, calibrated data or orientation data

Filter messages (Section 3.3.7)

Message	MID	Direction	Description
ReqHeading	130 (0x82)	To MTi/MTx	Request heading settings
ReqHeadingAck	131 (0x83)	To host	Device returns heading
SetHeading	130 (0x82)	To MTi/MTx	Host sets output mode
SetHeadingAck	131 (0x83)	To host	Device acknowledges SetHeading message
ReqInitTrackMode	136 (0x88)	To MTi/MTx	Request initial tracking mode
ReqInitTrackModeAck	137 (0x89)	To host	Device returns initial tracking mode
SetInitTrackMode	136 (0x88)	To MTi/MTx	Host sets initial tracking mode
SetInitTrackModeAck	137 (0x89)	To host	Device acknowledges SetInitTrackMode message
StoreFilterState	138 (0x8A)	To MTi/MTx	Stores gyro bias & mag bias in non-volatile memory
StoreFilterStateAck	139 (0x8B)	To host	Device acknowledges StoreFilterState message
ReqFilterParameters	160 (0xA0)	To MTi/MTx	Request a filter parameter used to calculate orientation
ReqFilterParametersAck	161 (0xA1)	To host	Device returns filter parameter
SetFilterParameters	160 (0xA0)	To MTi/MTx	Host sets a filter parameter
SetFilterParametersAck	161 (0xA1)	To host	Device acknowledges SetFilterParameters message
ReqAMD	162 (0xA2)	To MTi/MTx	Request adjust for magnetic disturbances or AMD mode
ReqAMDAck	163 (0xA3)	To host	Device returns the AMD mode
SetAMD	162 (0xA2)	To MTi/MTx	Host sets the AMD mode
SetAMDAck	163 (0xA3)	To host	Device acknowledges SetAMD message
ResetOrientation	164 (0xA4)	To MTi/MTx	Resets the orientation
ResetOrientationAck	165 (0xA5)	To host	Device acknowledges ResetOrientation message