Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

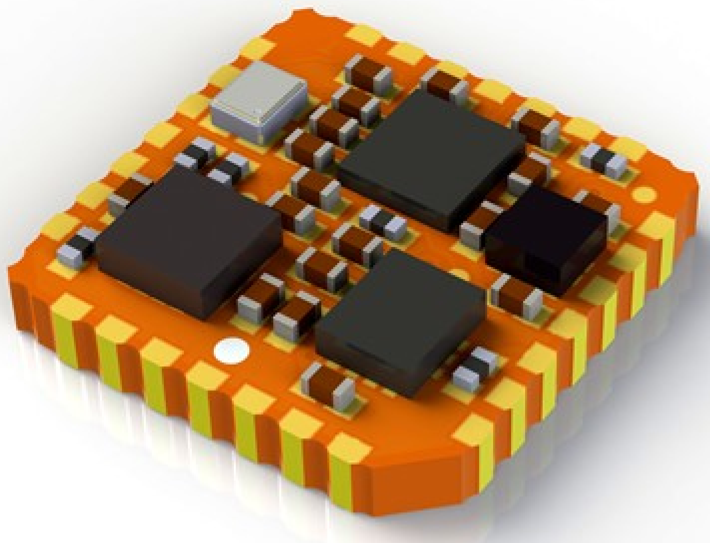
Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

MTi 1-series Data Sheet

IMU, VRU, AHRS and GNSS/INS module

Document MT0512P, Revision F, 1 July 2018



Xsens Technologies B.V.			Xsens North America, Inc.		
Pantheon 6a	phone	+31 (0)88 973 67 00	10557 Jefferson Blvd,	phone	310-481-1800
P.O. Box 559	fax	+31 (0)88 973 67 01	Suite C	fax	310-416-9044
7500 AN Enschede	e-mail	info@xsens.com	CA-90232 Culver City	e-mail	info@xsens.com
The Netherlands	internet	www.xsens.com	USA	internet	www.xsens.com

Revisions

Revision	Date	By	Changes
A	8 Jul 2015	MHA	Initial release
…	…	…	…
E	24 March 2018	AVY	Document caters only to - MTi-1 IMU (HW version ≥ 2.0) - MTi-2 VRU (HW version ≥ 2.0), orientation specification to be added in the next revision. - MTi-3 AHRS (HW version ≥ 2.0), orientation specification to be added in the next revision. - MTi-7 GNSS/INS (New release with HW version ≥ 2.0)
F	1 July 2018	AVY	Added MTi-2 VRU and MTi-3 AHRS performance specifications. Updated MTi-7 performance specifications.

Table of Contents

List of Tables

List of Figures

1. General information

This document provides information on the contents and usage of the MTi 1-series modules. The MTi 1-series module (MTi 1-s) is a fully functional, self-contained module that is easy to design-in. The MTi 1-s can be connected to a host through I^2C, SPI or UART interfaces.

The *Hardware Integration Manual: MTi 1-series* (MT1503) supplements this document. Notes on typical application scenarios, recommended external components, printed circuit board (PCB) layout, origin of measurements, stress related considerations, reference designs and handling information could be found in the hardware integration manual.

The *MT Low Level Communication Protocol* (MT0101P) document provides a complete reference for the protocol to communicate with Xsens Motion Trackers. For a better understanding of the Synchronous Serial Protocol (Section 3.4.3) for use with the MTi 1-s, the advice is to read the communication protocol reference in the *MT Low Level Communication Protocol* document first. The communication protocol document also describes the synchronization messages and settings in detail.

Links to the latest available documentation can be found in your MT Software Suite installation folder or via the following link: https://xsens.com/xsens-mti-documentation

1.1 Ordering information

Table 1: Ordering information for 1-series modules

Part Number	Output	Package	Packing Method
MTi-1T	IMU; inertial data	PCB, JEDEC-PLCC-28 compatible	Tray of 20
MTi-2T	VRU; inertial data, roll/pitch (referenced), yaw (unreferenced)	PCB, JEDEC-PLCC-28 compatible	Tray of 20
MTi-3T	AHRS; inertial data, roll/pitch/yaw	PCB, JEDEC-PLCC-28 compatible	Tray of 20
MTi-7T	GNSS/INS; inertial data, roll/pitch/yaw, velocity, position	PCB, JEDEC-PLCC-28 compatible	Tray of 20
MTi-1R	IMU; inertial data	PCB, JEDEC-PLCC-28 compatible	Reel of 250
MTi-2R	VRU; inertial data, roll/pitch (referenced), yaw (unreferenced)	PCB, JEDEC-PLCC-28 compatible	Reel of 250
MTi-3R	AHRS; inertial data, roll/pitch/yaw	PCB, JEDEC-PLCC-28 compatible	Reel of 250
MTi-7R	GNSS/INS; inertial data, roll/pitch/yaw, velocity, position	PCB, JEDEC-PLCC-28 compatible	Reel of 250

1.2 *Identifying device functionality using the unique Device Identifier*

Each Xsens product is marked with a unique serial device identifier referred to as the DeviceID. The DeviceID is categorized per MTi product configuration in order to make it possible to recognize the MTi by reviewing the DeviceID. The second digit of the DeviceID denotes the functionality (e.g. ‘1’ for MTi-1, MTi-10 and MTi-100), the third digit denotes the product series (6 for MTi 10-series, 7 for MTi 100-series, 8 for MTi 1-series) and the fourth digit denotes the interface (always 8 for the MTi 1-series). The last four digits are unique for each device; these four digits have a hexadecimal format.

Below is a list of the product types with their associated DeviceIDs. This rest of this document only caters to MTi 1-series with HW version ≥ 2.0. Refer to version MTi 1-series Datasheet rev D (5 Dec 2016) when you have an MTi with HW version < 2.0.

Table 2: Device ID's for MTi 1-series

Product (MTi 1-series)	DeviceID	HW version
MTi-1 IMU	0188xxxx	Check component layout (Section 6.4)
MTi-2 VRU	0288xxxx	
MTi-3 AHRS	0388xxxx	
MTi-7 GNSS/INS	0788xxxx	≥ 2.0 only

1.3 Block diagram

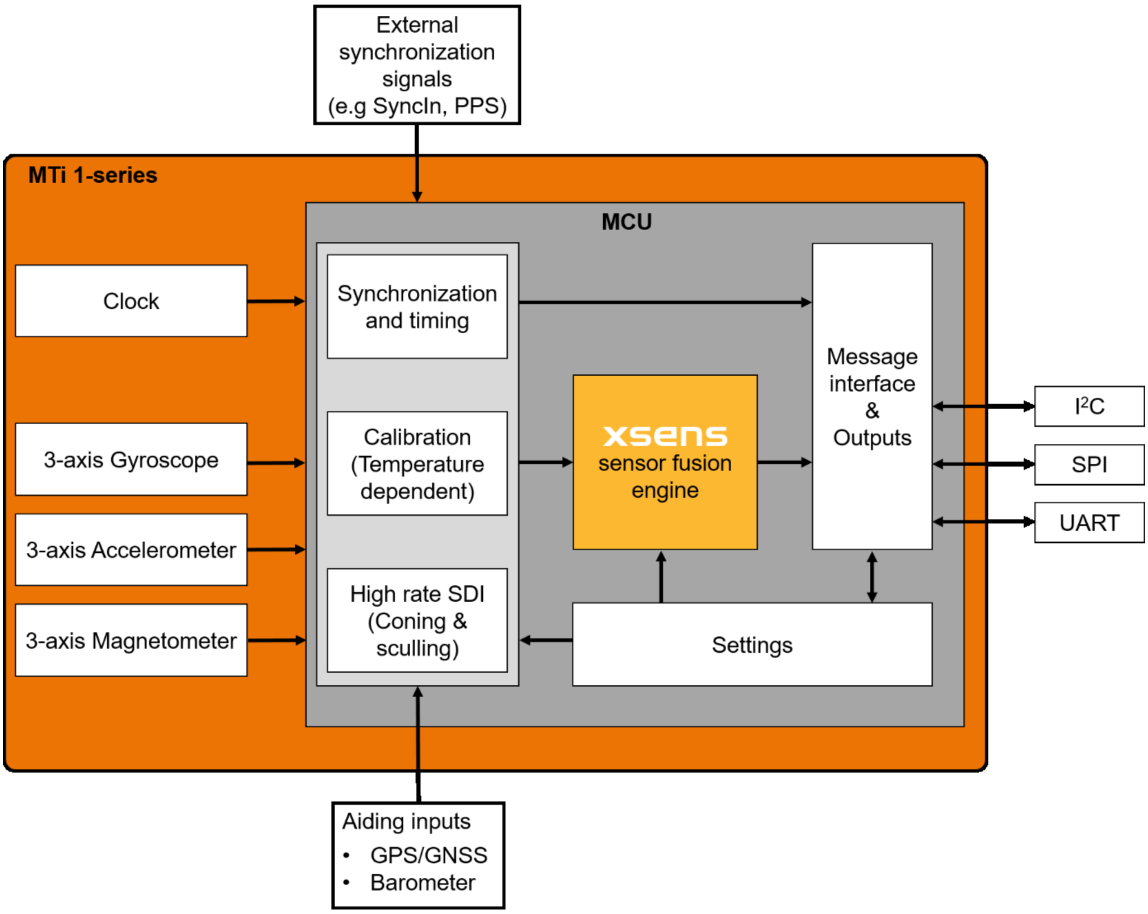


Figure 1: MTi 1-series module diagram

The diagram in Figure 1 shows a simplified organization of the MTi 1-series module (MTi 1-s). The MTi 1-s contains a 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer, a high-accuracy crystal and a low-power micro controller unit (MCU). The MTi-7 module can also accept the signals from an external GNSS receiver and/or barometer. The MCU coordinates the timing and synchronization of the various sensors. The module offers the possibility to use external signals in order to accurately synchronize the Mti 1-s with any user application. The MCU applies calibration models (unique to each sensor and including orientation, gain and bias offsets, plus more advanced relationships such as non-linear temperature effects and other higher order terms) and runs the Xsens optimized strapdown algorithm, which performs high-rate dead-reckoning calculations at 800 Hz, allowing accurate capture of high frequency motions and coning & sculling compensation. The Xsens sensor fusion engine combines all sensor inputs and optimally estimates the orientation, position and velocity at an output data rate of up to 100 Hz. The MTi 1-s is easily configurable for the outputs and depending on the application's needs can be set to use one of the filter profiles available within the Xsens sensor fusion engine. In this way, the MTi 1-s limits the load and the power consumption on the user application processor. The user can communicate with the module by means of three different communication interfaces. They are I2C, SPI and UART.

2. Sensor specifications

This section presents the performance and the sensor component specifications for the calibrated MTi 1-s module. Each module goes through the Xsens calibration process individually. The calibration procedure calibrates for many parameters, including bias (offset), alignment of the sensors with respect to the module PCB and each other and gain (scale factor). All calibration values are temperature dependent and temperature calibrated. The calibration values are stored in non-volatile memory of the module.

2.1 *MTi 1-series performance specifications*

Table 3: Orientation performance specifications

Parameter		MTi-1 IMU	MTi-2 VRU	MTi-3 AHRS	MTi-7 GNSS/INS
Roll/Pitch	Static	N/A	0.5º	0.5º	0.5º
	Dynamic	N/A	0.8º	0.8º	0.5º
Yaw	Dynamic	N/A	Unreferenced	2º	1.5º

Table 4: Position and velocity performance specifications (with MTi-7-DK)

Parameter		Specification
Position	Horizontal (SBAS)	1.0 m (1σ STD)
	Vertical (SBAS, baro)	2.0 m (1σ STD)
Velocity		0.05 m/s (1σ RMS)

All above specifications are based on typical application scenarios. The specifications mentioned in Table 3 and Table 4 are with MTi-3-DK and MTi-7-DK reference designs.

2.2 *Sensors specifications*

Table 5: Gyroscope specifications

Gyroscope specification[1]	Unit	MTi 1-series
Standard full range	[°/s]	±2000
In-run bias stability	[°/h]	10
Bandwidth (-3dB)	[Hz]	255
Noise density	[°/s/√Hz]	0.007
g-sensitivity (calibrated)	[°/s/g]	0.001
Sensitivity variation	[%]	0.05
Non-linearity	[%FS]	0.1

[1] As Xsens continues to update the sensors on the module, these specifications may change

Table 6: Accelerometer specifications

Accelerometer[2]	Unit	MTi 1-series
Standard full range	[g]	±16
In-run bias stability	[mg]	0.03
Bandwidth (-3dB)	[Hz]	324 (Z: 262)
Noise density	[µg/√Hz]	120
Sensitivity variation	[%]	0.05
Non-linearity	[%FS]	0.5

Table 7: Magnetometer specifications

Magnetometer[2]	Unit	MTi 1-series
Standard full range	[G]	8
Non-linearity	[%]	0.2
Total RMS noise	[mG]	0.5
Resolution	[mG]	0.25

Table 8: Alignment specifications

Parameter[2]	Unit	MTi 1-series
Non-orthogonality (accelerometer)	[°]	0.05
Non-orthogonality (gyroscope)	[°]	0.05
Non-orthogonality (magnetometer)	[°]	0.05
Alignment (gyr to acc)	[°]	0.05
Alignment (mag to acc)	[°]	0.1
Alignment of acc to the module board	[°]	0.2

[2] As Xsens continues to update the sensors on the module, these specifications may change

3. Functional description

This chapter describes the MTi 1-s pinout and gives details about the supported communication interfaces.

3.1 Pin configuration

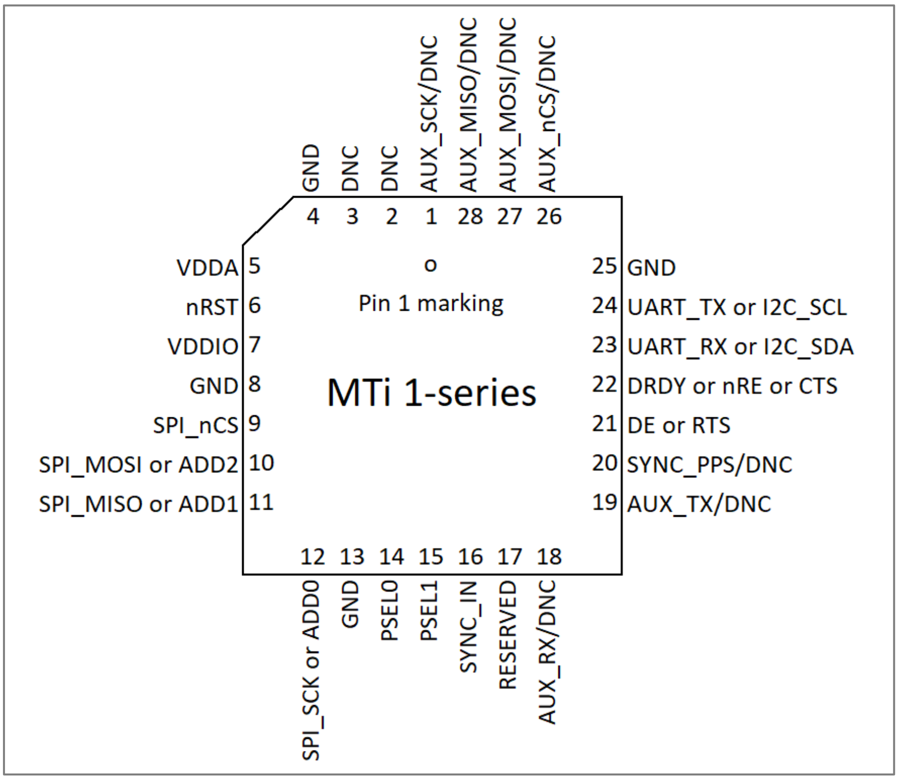


Figure 2: Pin configuration of the MTi 1-series module (top view)

3.2 Pin map

The pin map depends on the peripheral selection. See Section 3.4 on how to set the peripherals.

Table 9: Pin mapping for peripheral selection

	PSEL: I²C	PSEL: SPI	PSEL: UART half duplex	PSEL: UART full duplex
1	AUX_SCK/DNC[3]	AUX_SCK/DNC	AUX_SCK/DNC	AUX_SCK/DNC
2	DNC	DNC	DNC	DNC
3	DNC	DNC	DNC	DNC
4	GND	GND	GND	GND
5	VDDA	VDDA	VDDA	VDDA
6	nRST	nRST	nRST	nRST
7	VDDIO	VDDIO	VDDIO	VDDIO
8	GND	GND	GND	GND
9	DNC	SPI_nCS	DNC	DNC
10	ADD2[4]	SPI_MOSI	DNC	DNC
11	ADD1	SPI_MISO	DNC	DNC
12	ADD0	SPI_SCK	DNC	DNC
13	GND	GND	GND	GND
14	PSEL0	PSEL0	PSEL0	PSEL0
15	PSEL1	PSEL1	PSEL1	PSEL1
16	SYNC_IN	SYNC_IN	SYNC_IN	SYNC_IN
17	RESERVED	RESERVED	RESERVED	RESERVED
18	AUX_RX/DNC	AUX_RX/DNC	AUX_RX/DNC	AUX_RX/DNC
19	AUX_TX/DNC	AUX_TX/DNC	AUX_TX/DNC	AUX_TX/DNC
20	SYNC_PPS/DNC	SYNC_PPS/DNC	SYNC_PPS/DNC	SYNC_PPS/DNC
21	DNC	DNC	DE	RTS
22	DRDY	DRDY	nRE	CTS[5]
23	I2C_SDA	DNC	UART_RX	UART_RX
24	I2C_SCL	DNC	UART_TX	UART_TX
25	GND	GND	GND	GND
26	AUX_nCS/DNC	AUX_nCS/DNC	AUX_nCS/DNC	AUX_nCS/DNC
27	AUX_MOSI/DNC	AUX_MOSI/DNC	AUX_MOSI/DNC	AUX_MOSI/DNC
28	AUX_MISO/DNC	AUX_MISO/DNC	AUX_MISO/DNC	AUX_MISO/DNC

[3] AUX and SYNC_PPS pins are only available on MTi-7
[4] I²C addresses, see Table 13.
[5] CTS cannot be left unconnected if the interface is set to UART full duplex. If HW flow control is not used, connect o GND.

3.3 Pin descriptions

Table 10: Pin description MTi 1-series module

Name	Type	Description
Power Interface		
VDDA	Power	Analog power supply voltage for sensing elements
VDDIO	Power	Digital supply voltage. Also used as I/O reference.
Controls		
PSEL0	Selection pins	These pins determine the signal interface. See Table 11. Note that when the PSEL0/PSEL1 is not connected, its value is 1. When PSEL0/PSEL1 is connected to GND, its value is 0.
PSEL1		
nRST		Active low reset pin. Only drive with an open drain output or momentary (tactile) switch to GND. During normal operation this pin must be left floating, because this line is also used for internal resets. This pin has an internal weak pull-up to VDDIO.
ADD2	Selection pins	I^2C address selection lines. See Table 13 for list of I^2C addresses.
ADD1		
ADD0		
Signal Interface		
I2C_SDA	I^2C interface	I^2C serial data input/output
I2C_SCL		I^2C serial clock input
SPI_nCS	SPI interface	SPI chip select input (active low)
SPI_MOSI		SPI serial data input (slave)
SPI_MISO		SPI serial data output (slave)
SPI_SCK		SPI serial clock input
RTS	UART interface	Hardware flow control output in UART full duplex mode (Ready-to-Send)
CTS		Hardware flow control input in UART full duplex mode (Clear-to-Send). If flow control is not used connect to GND
nRE		Receiver control signal output in UART half duplex mode
DE		Transmitter control signal output in UART half duplex mode
UART_RX		Receiver data input
UART_TX		Transmitter data output
SYNC_IN	Sync interface	Accepts a trigger input to request the latest available data message
DRDY	Data ready	Data ready output indicates that data is available (SPI / I^2C)
Auxiliary interface (MTi-7 only)		
AUX_RX	Auxiliary GNSS interface	Receiver data input from GNSS module
AUX_TX		Transmitter data output to GNSS module
SYNC_PPS		Pulse per second input from GNSS module
AUX_nCS	Auxiliary SPI interface	SPI chip select output
AUX_MOSI		SPI serial data output (master)
AUX_MISO		SPI serial data input (master)
AUX_SCK		SPI serial clock output

3.4 *Peripheral interface selection*

The MTi 1-series module supports UART, I²C, and SPI interfaces for host communication. The host can select the active interface through the peripheral selection pins PSEL0 and PSEL1. The module reads the state of these pins at start-up, and configures its peripheral interface according Table 11. To change the selected interfaces, the host must first set the desired state of the PSEL0 and PSEL1 pins, and then reset the module. The module has internal pull-ups on the PSEL0 and PSEL1 pins. If these pins are left unconnected, the peripheral interface selection defaults to I²C (PSEL0 = 1, and PSEL1 = 1).

Table 11: Peripheral interface selection

Interface	PSEL1	PSEL0
I²C	1	1
SPI	1	0
UART half-duplex	0	1
UART full-duplex	0	0

3.4.1 Peripheral interface architecture

At its core, the module uses the Xsens-proprietary Xbus protocol which is compatible with all Xsens Motion Tracker products. This protocol is available on all interfaces, UART (asynchronous serial port interfaces), I²C and SPI interfaces. The I²C and SPI interfaces differ from UART in that they are synchronous, and have a master-slave relation in which the slave cannot send data by itself. This makes the Xbus protocol not directly transferable to these interfaces. For this purpose, the module introduces the MTi Synchronous Serial Protocol (MTSSP) protocol, a protocol for exchanging standard Xbus protocol messages over the I²C and SPI interfaces. Figure 3 shows how MTSSP fits in the module's (simplified) communication architecture. The module has generic Input- and Output-Queues for Xbus protocol messages. For I²C and SPI, the MTSSP layer translates these messages, while for the UART connection, the module transports the messages as-is.

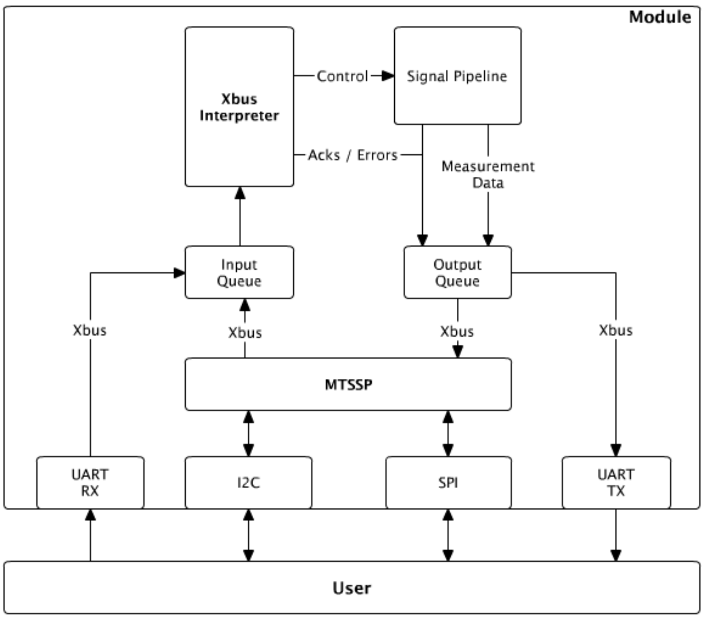


Figure 3: Communication architecture of MTi 1-series module (simplified)

3.4.2 Xbus Protocol

The Xbus protocol is Xsens' proprietary protocol for interfacing with the MTi 1-series. The MT Low Level Communication Protocol Documentation is a complete reference for the protocol. For a better understanding of the MTSSP explanation, the advice is to read the protocol reference first.

3.4.3 MTSSP Synchronous Serial Protocol

This Section specifies the MTi Synchronous Serial Protocol (MTSSP). The MTi 1-series module uses MTSSP as the communication protocol for both the I²C and SPI interfaces. The ARM® mbed™ example program (see https://developer.mbed.org/teams/Xsens), provides a reference implementation for the host side of the protocol.

Data flow

MTSSP communication happens according the master-slave model. The MTi 1-series module will always fulfill the slave-role while the user/integrator/host of the module is always the Master. The Master always initiates and drives communication. The Master either writes a message to the module, or reads a message from the module.

Figure 4 shows the data flow between the host (Master Device), and the MTi 1-s (Slave). The Master can control the Module by sending Xbus messages to the Control Pipe. The Module considers the bytes received in a single bus transfer to be exactly one Xbus message. The modules places the received message in the Input Queue for further handling. The Xbus Interpreter handles the messages in the Input Queue, and places the response messages in the Output Queue. The Master Device can read these response messages from the Notification Pipe.

The Master can switch the Module between configuration and measurement mode by sending the usual GotoConfig and GotoMeasurement messages to the Control Pipe. When placed in Measurement mode, the module will place the generated measurement data in the Measurement Pipe. The Master Device has to read the Measurement Pipe to received measurement data.

For the Master to know the size of the messages in the Notification- and Measurement pipes it can read the Pipe Status. The Pipe Status contains the size in bytes of the next message in both the Notification- and Measurement pipe. The Master can tweak the behavior of the protocol by writing the Protocol Configuration. The Master can also read the Protocol Configuration to check current behavior, and get protocol information.

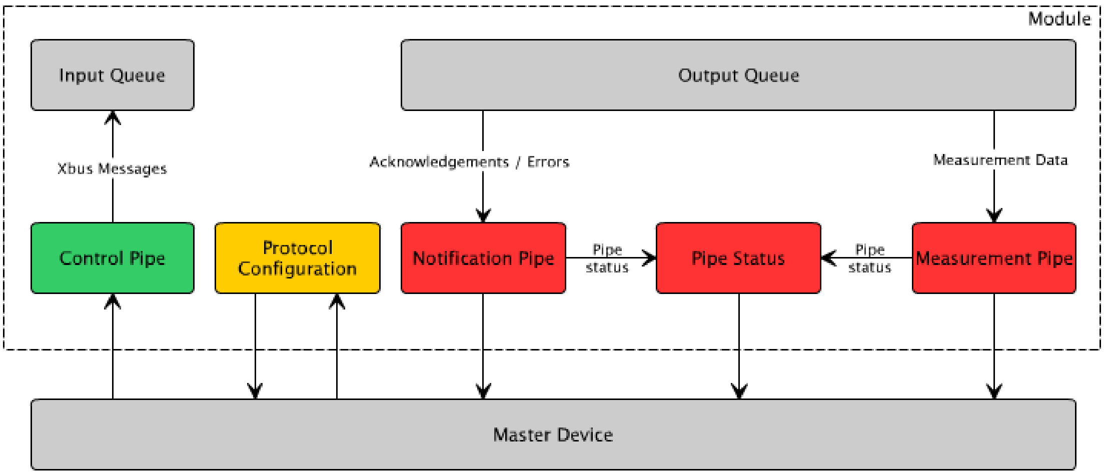


Figure 4: Data flows within MTSSP

Data ready signal
The Data Ready Signal (DRDY) is a notification line driven by the module. Its default behavior is to indicate the availability of new data in either the Notification- or the Measurement pipe. By default, the line is low when both pipes are empty, and will go high when either pipe contains an item. As soon as the Master has read all pending messages, the DRDY line will go low again.

The Master can change the behaviour of the DRDY signal using the Procotol Configuration. Please refer to the description of the ConfigureProtocol opcode (Table 12)for more information.

Opcodes
The Master starts each transfer with an opcode. The opcode determines the type of the transfer. The defined opcodes are as listed in Table 12. Following the opcode, and depending on whether it is a read- or write transfer, the Master either reads or writes one or more data bytes. The specific transfer format is dependent of the underlying bus protocol (I^2C or SPI), and is specified in Sections 3.4.4 and 3.4.5.

For some opcodes, the MTSSP uses reduced Xbus messages. A reduced Xbus message is a regular Xbus message with the preamble and busID fields removed to save bandwidth. These fields correspond to the first two bytes of a regular Xbus message. The calculation of the checksum for a reduced Xbus message still includes the busID and assumes its value to be 0xFF.

Table 12: List of defined opcodes

Opcode	Name	Read/Write	Data format	Description
0x01	ProtocolInfo	Read	Opcode defined	Status of the protocol behaviour, protocol version
0x02	ConfigureProtocol	Write	Opcode defined	Tweak the Protocol, e.g. the behaviour of the DRDY pin, behaviour of the pipes
0x03	ControlPipe	Write	Reduced Xbus	Used to send control messages to the module
0x04	PipeStatus	Read	Opcode defined	Provides status information for the read pipes
0x05	NotificationPipe	Read	Reduced Xbus	Used to read non-measurement data: errors acknowledgements and other notifications from the module
0x06	MeasurementPipe	Read	Reduced Xbus	All measurement data generated by the module will be available in the measurement pipe

ProtocolInfo (0x01)
The ProtocolInfo opcode allows the Master to read the active protocol configuration. The format of the message is as follows (All data is little endian, byte aligned):

```
struct MtsspInfo
{
    uint8_t m_version;
    uint8_t m_drdyConfig;
};
```

m_version:

7	6	5	4	3	2	1	0
VERSION [7:0]							

m_drdyConfig:

Bits 7:4	Reserved for future use
Bit 3	**MEVENT**: Measurement pipe DRDY event enable 0 : Generation of DRDY event is disabled 1 : Generation of DRDY event is enabled
Bit 2	**NEVENT**: Notification pipe DRDY event enable 0 : Generation of DRDY event is disabled 1 : Generation of DRDY event is enabled
Bit 1	**OTYPE**: Output type of DRDY pin 0: Push/pull 1: Open drain
Bit 0	**POL**: Polarity of DRDY signal 0: Idle low 1: Idle high

ConfigureProtocol (0x02)

The ProtocolInfo opcode allows the Master to change the active protocol configuration. The format of the message is as follows (All data is little endian, byte aligned):

```
struct MtsspConfiguration
{
    uint8_t m_drdyConfig;
};
```

m_drdyConfig:

Bits 7:4	Reserved for future use
Bit 3	**MEVENT**: Measurement pipe DRDY event enable 0 : Generation of DRDY event is disabled 1 : Generation of DRDY event is enabled
Bit 2	**NEVENT**: Notification pipe DRDY event enable 0 : Generation of DRDY event is disabled 1 : Generation of DRDY event is enabled
Bit 1	**OTYPE**: Output type of DRDY pin 0: Push/pull

	1: Open drain
Bit 0	**POL**: Polarity of DRDY signal 0: Idle high 1: Idle low

ControlPipe (0x03)
The ControlPipe opcode allows the Master to write messages to the Control Pipe. The bytes following the opcode represent a single (reduced) Xbus message

PipeStatus (0x04)
The PipeStatus opcode allows the Master to retrieve the status of the module's Notification- and Measurement pipes. The format of the message is as follows (All data is little endian, byte aligned):

```
struct MtsspConfiguration
{
    uint16_t m_notificationMessageSize;
    uint16_t m_measurementMessageSize;
};
```

NotificationPipe (0x05)
The Master uses the NotificationPipe opcode to read from the Notification Pipe. The read data is a single reduced Xbus message

MeasurementPipe (0x06)
The Master uses the MeasurementPipe opcode to read from the Measurement Pipe. The read data is a single reduced Xbus message

3.4.4 I²C

The MTi 1-series supports the I²C transport layer as of firmware 1.0.6. Note, it is not possible to upgrade devices with firmware revision 1.0.3 or earlier, to support this protocol. The MTi 1-series module acts as an I²C Slave. The User of the MTi 1-series module is the I²C Master.

The User can configure the I²C slave address through the ADD0, ADD1 and ADD2 pins. The module reads the state of these pins at start-up, and configures the slave address according to Table 13. The ADD0, ADD1 and ADD2 pins are pulled-up internally so when left unconnected the address selection defaults to 0x6B (ADD = 111).

Table 13: List of I²C addresses

I2C address	ADD2	ADD1	ADD0
0x1D	0	0	0
0x1E	0	0	1
0x28	0	1	0
0x29	0	1	1
0x68	1	0	0
0x69	1	0	1
0x6A	1	1	0
0x6B *(default)*	1	1	1

Table 14: Implemented I²C bus protocol features

Feature	Slave Requirement	MTi 1-series
7-bit slave address	Mandatory	Yes
10-bit slave address	Optional	No
Acknowledge	Mandatory	Yes
Arbitration	N/A	N/A
Clock stretching	Optional	Yes[6]
Device ID	Optional	No
General Call address	Optional	No
Software Reset	Optional	No
START byte	N/A	N/A
START condition	Mandatory	Yes
STOP condition	Mandatory	Yes
Synchronization	N/A	N/A

Writing to the MTi 1-s
Write operations consists of a single I²C write transfer. The Master addresses the module and the first byte it sends is the opcode. The bytes that follow are the data bytes. The interpretation of these data bytes depends on the opcode, as described in Section 3.4.3.

The maximum message size a module can receive is 512 bytes. If the Master sends more than 512 bytes, the module will reset its receive-buffer, which reduces the received message to consist only of the excess bytes.

Figure 5 shows the I²C transfer of a write message operation:

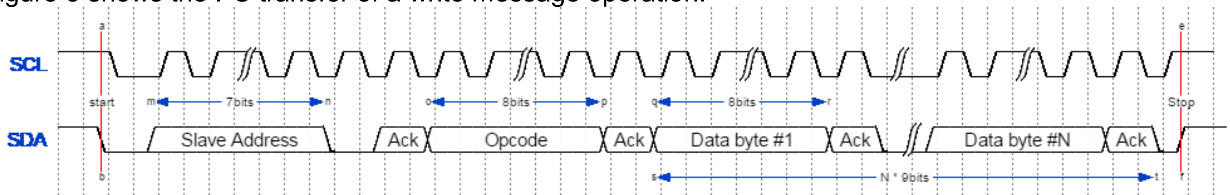


Figure 5: I²C write message operation

Reading from the module
To read from the module, the Master first does an I²C write transfer to transmit the opcode. The opcode tells the module what data the Master want to read. The module then prepares the requested data for transmission. The Master then does an I²C read transfer to retrieve the data. Figure 6 shows the I²C transfers for the described read method.

[6] The MTi-1 module relies on the I²C clock stretching feature to overcome fluctuations in processing time, the Master is required to support this feature

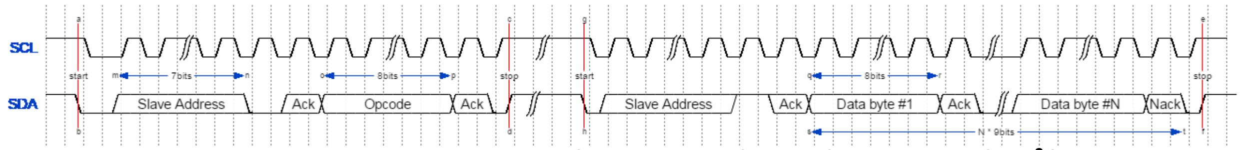


Figure 6: Read message operation with full write transfer and full read transfer (I²C)

Alternatively, the User can perform the read operation using a I²C transfer with a repeated start condition. Figure 7 depicts this read method.

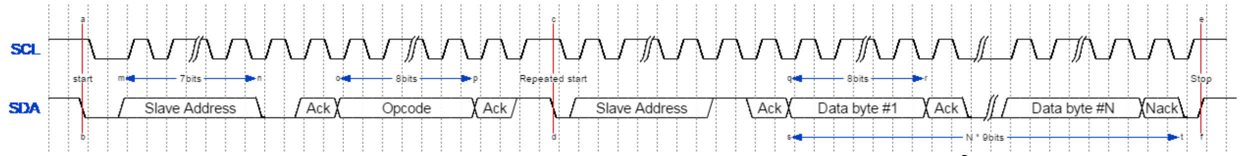


Figure 7: Read message transfer using a repeated start condition (I²C)

The Master controls how many data bytes it reads. For reading the Notification- and Measurement Pipes, the number of bytes the Master must read depends on the size of the pending message. In order to determine the correct number of bytes, the Master should first read the Pipe Status to obtain the sizes of the pending messages.

If the Master reads more bytes than necessary, the module will restart sending the requested data from the beginning.

3.4.5 SPI

The MTi 1-series supports the SPI transport layer as of firmware 1.0.6. Note, that it is not possible to upgrade devices with firmware revision 1.0.3 or earlier, to support this protocol.

The MTi 1-series module acts as an SPI Slave. The User of the MTi 1-series module is the SPI Master.

SPI Configuration

The MTi 1-series supports 4-wire mode SPI. The four lines used are:

- Chip Select (SPI_nCS)
- Serial Clock (SPI_SCK)
- Master data in, slave data out (SPI_MISO)
- Master data out, slave data in (SPI_MOSI)

The module uses SPI mode 3: It captures data on the rising clock edge, and it latches/propagates data on the falling clock edge. (CPOL=1 and CPHA=1).Data is clocked-out MSB first. The module uses an 8-bit data format.

Data transfer

The module uses a single type of SPI transfer for all communications. Figure 8 depicts this basic transfer.

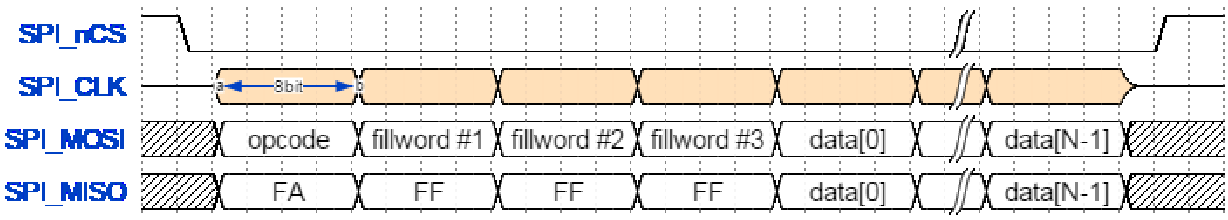


Figure 8: SPI basic transfer

The Master starts a transfer by pulling the SPI_nCS low, in order to select the Slave. The Master must keep the SPI_nCS line low for the duration of the transfer. The Slave will interpret the rising edge of the SPI_nCS line as the end of the transfer. The Master places the data it needs to transmit on the SPI_MOSI line. The Slave will place its data on the SPI_MISO line.

The Master first transmits the opcode. The opcode determines what kind of data the Master transmits, and what kind of data the Master wants to read from the Slave (See Section 3.4.3). The second- to fourth byte the Master transmits are the fill words. These fill words are needed to give the Slave time to select the data it must send in the remainder of the transfer. Both Master and Slave are free to choose the value of the fill words, and the receiving end should ignore their value. However, the first 4 bytes transmitted by the MTi 1-series module (Slave) are always 0xFA, 0xFF, 0xFF, 0xFF.

Following the first four words are the actual data of the transfer. It is the responsibility of the Master to determine how many bytes it must transfer. For reading the Notification- and Measurement Pipes, the number of bytes the Master must read depends on the size of the pending message. In order to determine the correct number of bytes, the Master should first read the Pipe Status to obtain the sizes of the pending messages.

Timing
Table 15 and Figure 9 specify the timing constraints that apply to the SPI transport layer. The Master must adhere to these constraints.

Table 15: SPI timing

Symbol	Parameter	Unit	Min	Max
T1	Slave select to first complete word delay	µs	4	
T2	Byte time	µs	4	
T3	Consecutive SPI transfer guard time	µs	3	
	Max SPI bitrate	Mbit		2

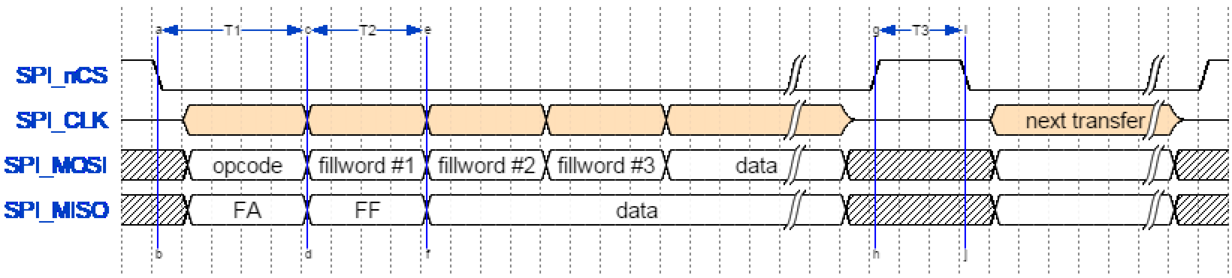


Figure 9: SPI timing

3.4.6 UART half-duplex

The User can configure the MTi 1-series module to communicate over UART in half-duplex mode. The UART frame configuration is 8 data bits, no parity and 1 stop bit (8N1). In addition to the RX and TX pins, the modules uses control lines nRE and DE. The modules uses these control outputs to drive the TX signal on a shared medium and to drive the signal of the shared medium on the RX signal.

A typical use case for this mode is to control an RS485 transceiver where the shared medium is the RS485 signal, and the nRE and DE lines control the buffers inside the transceiver.

When the module is transmitting data on its TX pin it will raise both the nRE and DE lines, else it will pull these lines low. Figure 10 depicts the behaviour of the involved signals.

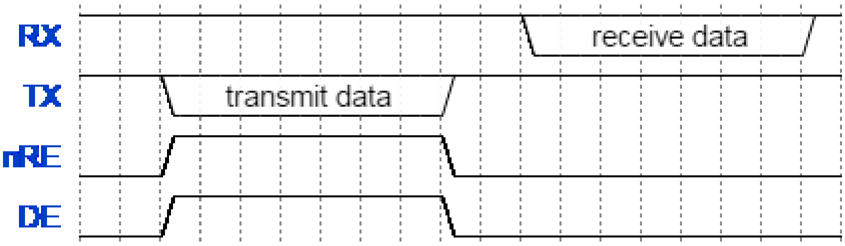


Figure 10: Behaviour of the nRE and DE lines

Note that in this mode the UART of the MTi 1-s itself is still operating full duplex.

3.4.7 UART full duplex with RTS/CTS flow control

The user can configure the MTi 1-s module to communicate over UART in full duplex mode with RTS/CTS flow control. The UART frame configuration is 8 data bits, no parity and 1 stop bit (8N1). In addition to the RX and TX signals for data communication, the module uses the RTS and CTS signals for hardware flow control.

The CTS signal is an input for the module. The module checks the state of the CTS line at the start of every byte it transmits. If CTS is low, the module transmits the byte. Otherwise, it postpones transmission until CTS is lowered. When during the transmission of a byte the User raises the CTS signal, then the module completes transmission of that byte before postponing further output. The module will not retransmit this byte. Figure 11 shows the behaviour of the TX and CTS lines.

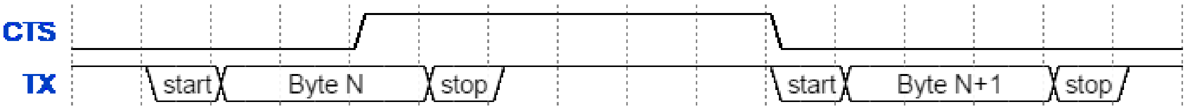


Figure 11: Data transmit behaviour under CTS

The RTS signal is an output for the module. If the RTS line is high, the module is busy and unable to receive new data. Otherwise, the module's UART is idle and ready to receive. After receiving a byte the direct memory access (DMA) controller of the module will transfer the byte to its receive first in first out (FIFO) buffer. The module will raise the RTS signal during this transfer. Therefore, with every byte received, the module raises the RTS line shortly. Figure 12 shows this behaviour.

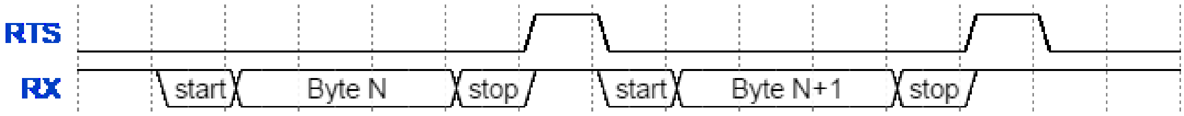


Figure 12: RTS behaviour under data reception

This User can use this communication mode without hardware flow control. In this case, the user must tie the CTS line low (GND) to make the module transmit.

4. MTi 1-series architecture

This section discusses the MTi 1-s module architecture including the various configurations available and the signal processing pipeline.

4.1 MTi 1-series configurations

The MTi 1-s module is a fully tested self-contained module available as an Inertial Measurement Unit (IMU), Vertical Reference Unit (VRU), Attitude and Heading Reference System (AHRS) and GNSS aided Inertial Navigation System (GNSS/INS). It can output 3D orientation data (Euler angles, rotation matrix or quaternions), orientation and velocity increments (Δq and Δv), position and velocity quantities and calibrated sensors data (acceleration, rate of turn, magnetic field). Depending on the product, output options may be limited to sensors data and/or unreferenced yaw.

The MTi 1-s module features a 3D accelerometer, a 3D gyroscope, a magnetometer, a high-accuracy crystal and a low-power MCU. The MCU coordinates the timing and synchronization of the various sensors, applies calibration models (e.g. temperature models) and runs the sensor fusion algorithm. The MCU also generates output messages according to the proprietary XBus communication protocol. The data output are fully configurable, so that the MTi 1-s limits the load, and thus power consumption, on the user application processor.

4.1.1 MTi-1 IMU

The MTi-1 module is an IMU that outputs calibrated 3D rate of turn, 3D acceleration and 3D magnetic field. The MTi-1 also outputs coning and sculling compensated orientation increments and velocity increments (Δq and Δv). Advantages over a gyroscope-accelerometer combo-sensor are the inclusion of synchronized magnetic field data, on-board signal processing and the easy-to-use synchronization and communication protocol. Moreover, the testing and calibration over temperature performed by Xsens result in a robust and reliable sensor module, that can be integrated within a short time frame. The signal processing pipeline and the suite of output options allow access to the highest possible accuracy at any output data rate, limiting the load on the user application processor.

4.1.2 MTi-2 VRU

The MTi-2 is a 3D VRU. Its algorithm computes 3D orientation data with respect to a gravity referenced frame: drift-free roll, pitch and unreferenced yaw. Although the yaw is unreferenced, it is superior to gyroscope integration. In addition, it outputs calibrated sensor data: 3D acceleration, 3D rate of turn and 3D magnetic field data. All modules of the MTi 1-series output data generated by the strapdown integration algorithm (orientation and velocity increments - Δq and Δv). The 3D acceleration is also available as so-called free acceleration, which has the local-gravity subtracted. The drift in unreferenced heading can be limited using the Active Heading Stabilization (AHS) feature, see Section 4.3.3 for more details.

4.1.3 MTi-3 AHRS

The MTi-3 supports all features of the MTi-1 and MTi-2, and in addition is a full magnetometer-enhanced AHRS. In addition to the roll and pitch, it outputs a true magnetic North referenced yaw and calibrated sensors data: 3D acceleration, 3D rate of turn, 3D orientation and velocity increments (Δq and Δv) and 3D earth-magnetic field data. Free acceleration is also computed by the MTi-3 AHRS.

4.1.4 MTi-7 GNSS/INS

The MTi-7 provides a GNSS/INS solution offering a position and velocity output in addition to orientation output. The MTi-7 uses advanced sensor fusion algorithms developed by Xsens to synchronize the inputs from the module’s on-board accelerometer, gyroscope and magnetometer with the data from an external GNSS receiver and/or barometer. The raw sensor signals are combined and processed at a high data rate of 800 Hz to produce a real-time data stream with device's 3D position, velocity and orientation (roll, pitch and yaw).

4.2 *Signal processing pipeline*

The MTi 1-series is a self-contained module, so all calculations and processes such as sampling, coning & sculling compensation and the Xsens sensor fusion algorithm run on board.

4.2.1 Strapdown integration

The Xsens optimized strapdown algorithm performs high-rate dead-reckoning calculations at 800 Hz allowing accurate capture of high frequency motions. This approach ensures a high bandwidth. Orientation and velocity increments are calculated with full coning & sculling compensation. These orientation and velocity increments are suitable for any 3D motion tracking algorithm. Increments are internally time-synchronized with other sensor. The output data rate can be configured with different frequencies up to 100 Hz. The inherent design of the signal pipeline with the computation of orientation and velocity increments ensure there is absolute no loss of information at any output data rate. This makes the MTi 1-series attractive for systems with limited communication bandwidth.

4.2.2 Xsens sensor fusion algorithm for VRU and AHRS product types

Xsens sensor fusion algorithm optimally estimates the orientation with respect to an Earth fixed frame utilizing the 3D inertial sensor data (orientation and velocity increments) and 3D magnetometer.

The user can set the sensor fusion algorithm with different filter profiles in order to get the best performance based on the application scenario (see Table 16). These filter profiles contain predefined filter parameter settings suitable for different user application scenarios.

In addition, all filter profiles can be used with the Active Heading Stabilization (AHS) setting, which significantly reduces heading drift during magnetic disturbances. The In-run Compass Calibration (ICC) setting can be used to compensate for magnetic distortions that are caused by every object the MTi is attached to. See Section 4.3 for more details.

Table 16: Filter profiles for VRU and AHRS

Name	Number	Product	Description
General	50	MTi-3	Suitable for most applications.
High_mag_dep	51	MTi-3	Heading corrections strongly rely on the magnetic field measured and should be used when magnetic field is homogeneous.
Dynamic	52	MTi-3	Assumes that the motion is highly dynamic.
North_referenced	53	MTi-3	Assumes a good Magnetic Field Mapping (MFM) and a homogeneous magnetic field. Given stable initialization procedures and observability of the gyro bias, after dynamics, this filter profile will trust more on the gyro solution and the heading will slowly converge to the disturbed mag field over the course of time.
VRU_general	54	MTi-2 MTi-3	Magnetometers are not used to determine heading. Consider using VRU_general in environments that have a heavily disturbed magnetic field or when the application only requires unreferenced heading (see also Section 4.3.3).

4.2.3 Xsens sensor fusion algorithm for GNSS/INS product

The Xsens sensor fusion algorithm in the MTi-7 has several advanced features. The MTi-7 algorithm adds robustness to the orientation and position estimates by combining measurements and estimates from the inertial sensors and GNSS receiver in order to compensate for transient accelerations and magnetic disturbances.

When the MTi-7 has limited/mediocre GNSS reception or even no GNSS reception at all (outage), the MTi-7 sensor fusion algorithm seamlessly adjusts the filter settings in such a way that the highest possible accuracy is maintained. The GNSS status is continuously monitored and the filter accepts GNSS data when available and sufficiently trustworthy. The sensor will continue to output position, velocity and orientation estimates, although the accuracy is likely to degrade over time as the filters will have to rely on dead-reckoning. If the GNSS outage lasts longer than 45 seconds, the MTi-7 stops output of the position and velocity estimates, and begins sending these outputs once the GNSS data becomes acceptable again.

Table 17 reports the different filter profiles the user can set based on the application scenario. Every application is different and results may vary from setup to setup. It is recommended to reprocess recorded data with different filter profiles in MT Manager to determine the best results in your specific application.

Table 17: Filter profiles for MTi-7 (GNSS/INS)

Name	Number	GNSS[7]	Barometer[7]	Magnetometer	Description
General	11	●	●		This filter profile is the default setting. Yaw is North referenced (when GNSS is available). Altitude (height) is determined by combining static pressure, GNSS altitude and accelerometers. The barometric baseline is referenced by GNSS, so during GNSS outages, accuracy of height measurements is maintained.
GeneralNoBaro	12	●			This filter profile is very similar to the general filter profile except for the use of barometer.
GeneralMag[8]	13	●	●	●	This filter profile bases its yaw estimate mainly on magnetic heading and GNSS measurements. A homogenous or magnetic field calibration is essential for good performance.

[7] External aiding sensors for the MTi-7
[8] This filter profile can be used even when the barometer is not part of the design.

4.3 Magnetic interference

Magnetic interference can be a major source of error for the heading accuracy of any AHRS. As an AHRS uses the magnetic field to reference the dead-reckoned orientation on the horizontal plane with respect to the (magnetic) North, a severe and prolonged distortion in that magnetic field will cause the magnetic reference to be inaccurate. The MTi 1-series module has several ways to cope with these distortions to minimize the effect on the estimated orientation.

4.3.1 Magnetic Field Mapping

When the distortion moves with the MTi (i.e. when a ferromagnetic object solidly moves with the MTi module), the MTi can be calibrated for this distortion. Examples are the cases where the MTi is attached to a car, aircraft, ship or other platforms that can distort the magnetic field. It also handles situations in which the sensor has become magnetized. These type of errors are usually referred to as soft and hard iron distortions. The Magnetic Field Mapping procedure compensates for both hard iron and soft iron distortions.

The magnetic field mapping (calibration) is performed by moving the MTi together with the object/platform that is causing the distortion. The results are processed on an external computer (Windows or Linux), and the updated magnetic field calibration values are written to the non-volatile memory of the MTi 1-series module. The magnetic field mapping procedure is extensively documented in the Magnetic Field Mapper User Manual (MT0202P), available in the MT Software Suite.

4.3.2 In-run Compass Calibration (ICC)

In-run Compass Calibration is a way to calibrate for magnetic distortions present in the sensor operation environment using an onboard algorithm. The ICC is an alternative for the offline MFM (Magnetic Field Mapper). It results in a solution that can run embedded on different industrial platforms (leaving out the need for a host processor like a PC) and relies less on specific user input. The MFM tool, which does require a host processor, is however still recommended over or in addition to the ICC. The ICC is aimed at applications for which the MFM solution cannot be used (e.g. MTi 1-s that is not able to be connected to a PC), when MFM is not sufficient (e.g. applications that move outside of the plane of motion used during the calibration), or when the user uses the same MFM result performed for one sensor to calibrate different sensors (typical for large volume applications).

It should be noted that magnetic distortions present in the environment of the motion tracker that move independently or change over time are not compensated by the ICC unless they are changing very slowly. Such distortions do not affect the parameter estimation; they are simply not compensated for. This also means that (ferromagnetic) objects should not be attached to or detached from the sensor while ICC is running.

If the user is able to perform a calibration motion in a homogeneous magnetic field or environment that is representative for the application, then it is possible to use "Representative Motion" feature (RepMo). RepMo is available in MT Manager, XDA and Low-Level Communication Protocol (Xbus protocol).

Additional examples are available in BASE: https://base.xsens.com/hc/en-us/articles/213588029.

4.3.3 Active Heading Stabilization (AHS)

The Active Heading Stabilization (AHS) is a software component within the sensor fusion engine designed to give low-drift unreferenced heading solution in a disturbed magnetic environment. AHS is available in all filter profiles. See MT Low Level Communication Protocol documentation for use of AHS feature.