# Modbus for SKF Multilog On-line System IMx and SKF @ptitude Observer



User Manual
Part Number **15V-090-00051-100**
Revision **E – April 2021**

Read this manual carefully before using the product. Failure to follow the instructions and safety precautions in this manual can result in serious injury, damage to the product or incorrect readings. Keep this manual in a safe location for future reference.



SKF Sverige AB
Aurorum 30, 977 75 Luleå, Sweden
Telephone: +46 (0) 31 337 10 00

® SKF is a registered trademark of the SKF Group.

Android platform is a trademark of Google LLC.

iOS is a registered trademark of Cisco in the US and is used by Apple under license.

Bluetooth is a registered trademark of Bluetooth SIG, Inc.

All other trademarks are the property of their respective owners.



Patents: US 4,768,380 • US 5,633,811 • US 5,679,900 • US 5,845,230 • US 5,852,351 • US 5,854,553 • US 5,854,994 • US 5,870,699 • US 5,907,491 • US 5,992,237 • US 6,006,164 • US 6,124,692 • US 6,138,078 • US 6,199,422 • US 6,202,491 • US 6,275,781 • US 6,301,514 • US 6,437,692 • US 6,489,884 • US 6,513,386 • US 6,633,822 • US 6,789,025 • US 6,792,360 • US 7,103,511 • US 7,697,492 • WO/2003/048714

### Product Registration

Please take a moment to register your product to receive exclusive benefits offered only to our registered customers, including technical support, tracking your proof of ownership and staying informed about upgrades and special offers. (Please visit our website for more details on these benefits.)

### General Product Information

General information such as datasheets and catalogues are published on the Condition Monitoring Systems site on SKF.com. Supporting product information can also be downloaded from the SKF Technical Support self-service web portal.

### Product Support Contact Information

***Repair** and **Calibration Services*** – Submit a Return Authorization (RA) request to arrange for repair or calibration of your product. You will receive an RA number and shipping instructions usually within 48 business hours.

***Product Support Plans (PSP)*** – SKF offers annual renewal Product Support Plans (PSP) on many condition monitoring products in an effort to extend the life of your product. Software and firmware updates are an exclusive entitlement to PSP customers. Additional benefits include product repair, Annual Preventative Maintenance (APM) and certified calibration - all of which are all carried out on a priority-basis. Enjoy unlimited technical support and access to after-hours support for machine- and process-critical applications.

***Product Sales*** – For information on purchasing condition monitoring products, services and support on products out of warranty, please contact your local SKF sales office or distributor.

*Technical Support* – SKF's Technical Support Group can be reached during normal business hours via phone, e-mail and live chat. Always check the self-service web portal before contacting your nearest Technical Support Group (TSG) to see if the answer is already published. You may search the vast knowledgebase within the self-service web portal for answers to commonly-asked questions (FAQ), how-to articles, technical specs, installation and user manuals, best practices and more.

Customers in Europe, Middle East and Africa:

- Phone: +46 (0) 31 337 6500
- E-Mail: TSG-EMEA@skf.com
- Chat: www.skf.com/cm/tsg

Customers in the Americas, Asia and all other locations:

- Phone: 1-858-496-3627 or toll-free (USA) 1-800-523-7514
- E-Mail: TSG-Americas@skf.com
- Chat: www.skf.com/cm/tsg

120517dm-fp-Feb_2020

# Table of contents

# 1 Introduction

## 1.1 Modbus for SKF Multilog IMx devices

Modbus is a communications protocol for connecting industrial electronic devices. SKF systems typically use Modbus for interfacing IMx-devices to control systems and IMx devices to other equipment, such as sensors.

The Modbus feature is configured through the SKF @ptitude Observer software or when using one of the IMx-8 / IMx-16 family as a stand-alone device, the SKF Multilog IMx Manager mobile application.

SKF Multilog IMx devices support the Modbus RTU protocol over 2-wire RS-485 as well as the Modbus TCP/IP protocol. Both allow for configuration as Modbus master or Modbus slave, devices.

The following are common scenarios of when Modbus functionalities are used with an IMx:

- To transmit data to a control system
- To receive data from a control system
- To receive data from sensors

Another Modbus interfacing possibility is a so called Direct Modbus device. This is a somewhat hybrid approach where the Direct Modbus device is a TCP/IP slave to an IMx which while taking up one Modbus connection, wouldn't use up any registers on that IMx. The IMx in this instance acts like a messaging relay between the software and the remote Modbus device. A typical application might use this to connect @ptitude Observer with an Oil Debris Monitor (sensor). For more information on the use of Direct Modbus devices refer to the @ptitude Observer User Manual.

## 1.2 About This User Manual

In this manual, certain formatting conventions are used:

**Bold** type is used to indicate text that appears on a menu, window or dialog.

➢ are used to indicate notes to the reader.

This user manual describes how to use Modbus with SKF Multilog IMx devices noting that:

➢ Information specific to the older and legacy IMx designs (IMx-S, IMx-W, IMx-C, IMx-B and IMx-T) is included only in Appendix B.

➢ Some functionality described may require latest @ptitude Observer software and IMx firmware. Regularly update the firmware held in @ptitude Observer to be able to take advantage of the latest features.

➢ Information on how to use Modbus with IMx-M devices is described in the "IMx-M User Manual", Rev O or later.

The document describes:

- How to physically configure Modbus/RTU protocol over RS-485, the cable connections and the Modbus (RS-485) interfaces on the various IMx devices.
- How to use Modbus capabilities – Modbus and IMx.
- How an IMx in the IMx-8 or IMx-16 family operating in stand-alone mode is configured for Modbus.
- Troubleshooting.

## 1.3 Related documentation

*Table 1 Related documentation*

| Document | Part No. |
|---|---|
| IMx-M User Manual, Rev O or later | 32179800 |
| IMx-8/IMx-8Plus User Manual, Rev E or later | 15V-090-00049-100 |
| IMx-16/IMx-16Plus User Manual, Rev C or later | 15V-090-00081-100 |
| IMx-Rail Supplementary User Manual, Rev A or later | 15V-090-00082-100 |
| SKF @ptitude Observer User Manual, Rev V or later | 32170900 |
| General Modbus protocol considerations for IMx devices | CM3229-EN |

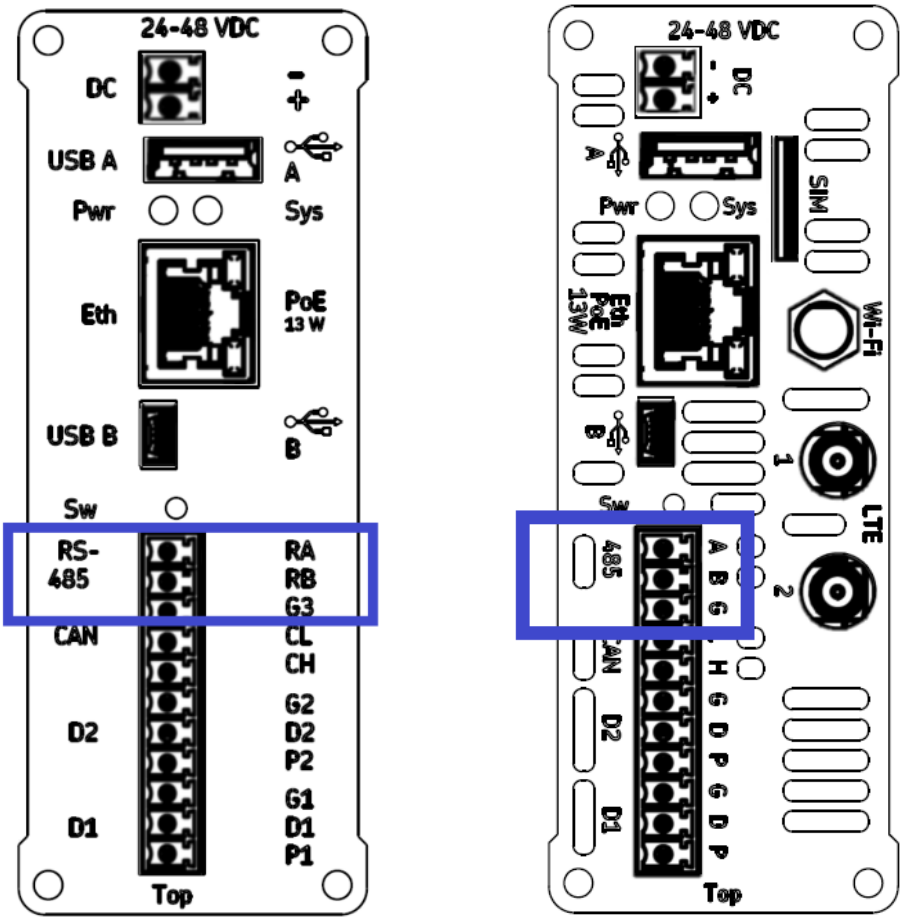
## 1.4 Important messages

*Important messages, instructions and information in this manual must be carefully followed. Otherwise, harm might occur to equipment and/or personnel.*

# 2 Modbus over RS-485

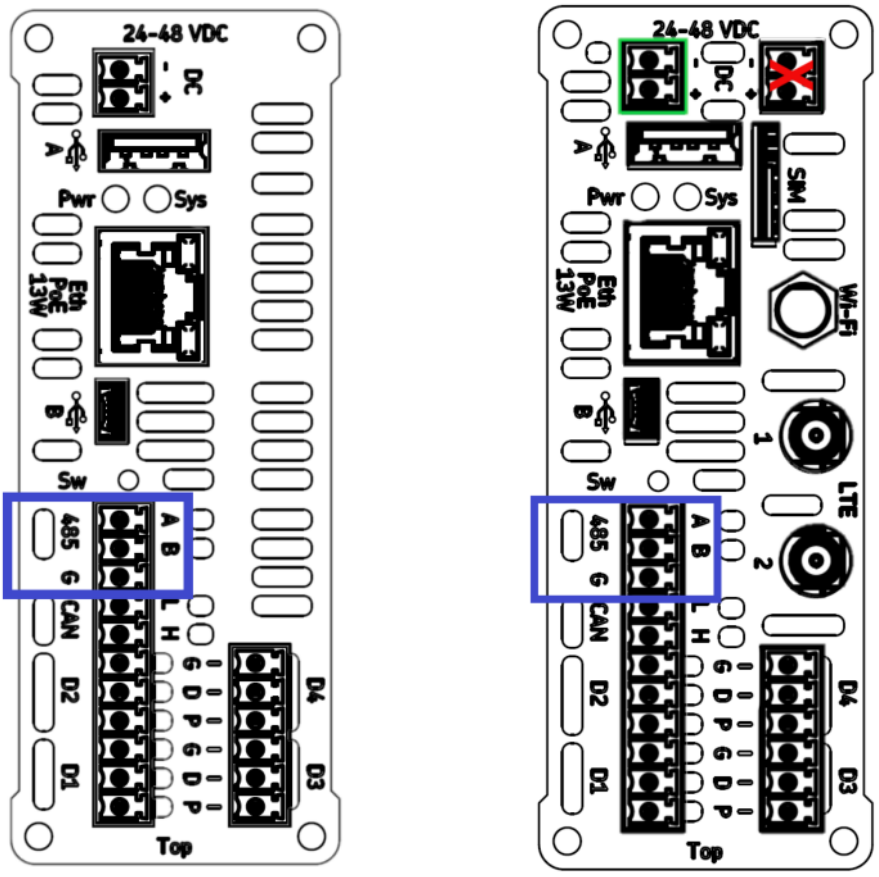
## 2.1 Terminal Blocks

When using Modbus RTU, connect to the RS-485 and when necessary, to the ground connections that are available at the terminal block of the IMx device. Some examples from IMx devices are shown in the figures below:




a) IMx-8 b) IMx-8Plus

*Figure 1 IMx-8 and IMx-8Plus RS-485 connections*




a) IMx-16 b) IMx-16Plus

*Figure 2 IMx-16 and IMx-16Plus RS-485 connections*

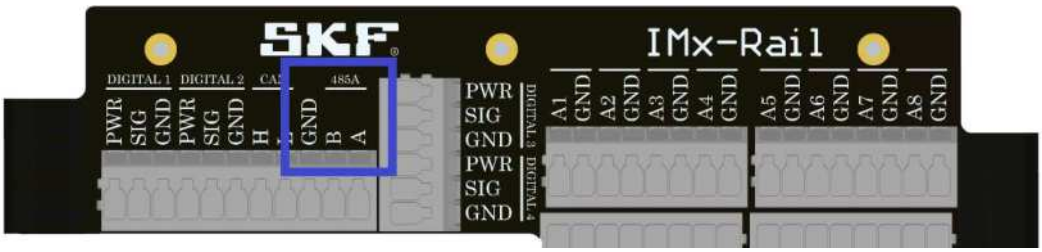



*Figure 3 IMx-Rail RS-485 terminals*

- A or RA = Non-inverting
- B or RB = Inverting
- GND, G or G3 = Reference ground

Note that some RS-485 equipment has the A and B naming reversed. In such cases, it will be necessary to connect the wires in reverse fashion to avoid communication failures (no damage can result, due to crossed A/B wiring).

Devices connected to the bus must have the same ground potential. If 'floating' equipment is to be connected to the RS-485 bus, the GND/G/G3 connection can be used to ensure the same ground potential.

## 2.2 Transmission Characteristics

The supported transmission parameters are listed in the table, below. The most commonly used values are marked with bold text.

Any combination of transfer/baud rate, parity and stop bits can be chosen. However, these parameters must match those of other equipment on the bus.

*Table 2 Supported RS-485 Transmission Parameters*

| RS-485 Transmission Parameter | Supported values |
|---|---|
| Transfer rate (bps) | 9600 / **19 200** |
| Parity | **None** / Even / Odd |
| Stop bits | **1** / 2 |

## 2.3 RS-485 Bus Termination

The IMx-8 and IMx-16 family of devices all have one, 2-wire, RS-485 port. For this there are considerations related to grounding and bus termination, **Figure 4**.

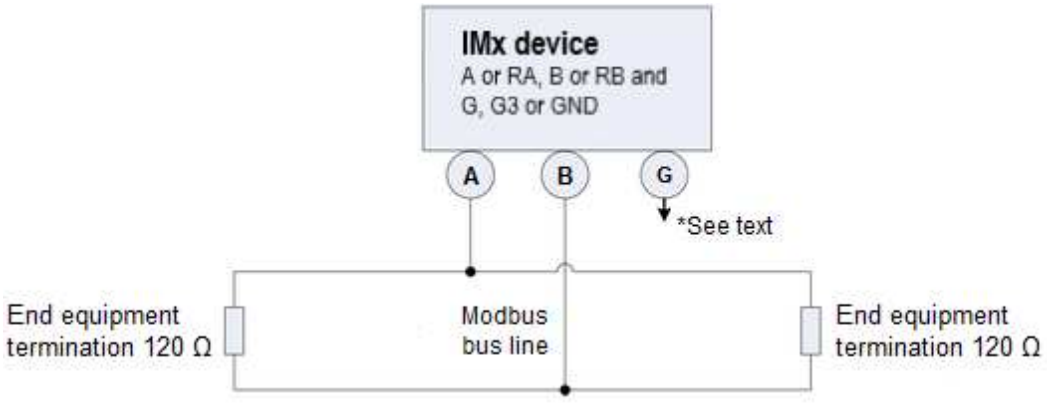



*Figure 4 Modbus Bus Connection and End Termination*

*G/G3/GND (Ground): Devices connected to the bus must have the same ground potential. When 'floating' equipment is connected to the IMx bus, the G/G3/GND connection can be used to ensure the same ground potential. Normally, all devices are connected to the same ground connection.

> *Important - To avoid ground loops, ensure that there is only one ground connection. The IMx device's G/G3/GND (Ground) connector can be used in cases where the connected equipment bus is floating.*

If the IMx device is placed first or last in the bus chain, then an external bus termination resistor must be connected to it.

- On the IMx-8 and IMx-16 family of devices there is no built-in termination that can be activated by configuration or by physical linking.

An external resistor of 120 ohms is provided by SKF as well as double deck terminals to facilitate its installation. This resistor is colour-coded black, **Figure 5**.

*Figure 5 120-ohm resistor for RS-485, bus termination*

Note that the IMx does not provide fail-safe bias to the bus.

# 3 Modbus TCP/IP

Modbus TCP/IP is a somewhat different communications protocol compared with Modbus RTU:

- Modbus TCP/IP does not require a CRC as the lower layers of the TCP/IP protocol stack already provide checksum protection.
- Communication across Ethernet uses the IP address rather than the slave address, but a 'slave address' is retained (as the 'Unit ID').

The difference between the two variants, regarding the structure of the messages used, is illustrated in the figure below.

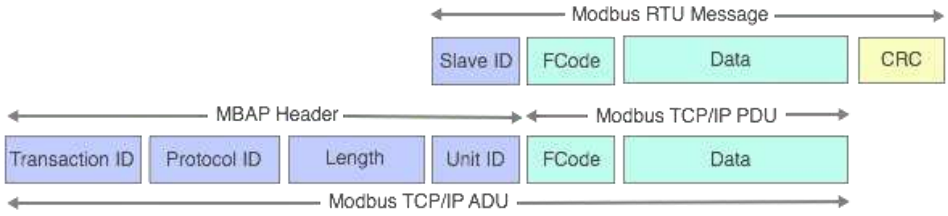



*Figure 6 Difference Between Modbus RTU and Modbus TCP/IP Messages*

In the IMx configuration settings in @ptitude Observer, for Modbus TCP/IP where the IMx is a slave, the TCP/IP address is hidden and the **Port** number is read only. For Modbus TCP/IP where the **Mode** is master, **IP address** and **Port** fields are available to enter the TCP/IP addresses of the desired target devices. Note that:

- The standard TCP server port is 502. This is the default in @ptitude Observer software and a fixed setting when the IMx is configured as a slave device.
- Apart from the communication settings, the configuration of measurement points to be exported and the setup for the external channels when the IMx device imports data either as a slave or as a master is the same for Modbus TCP/IP and Modbus RTU, as described in the Modbus and IMx and Configuring Modbus Capability for IMx, sections.
- Direct Modbus devices are configured as Modbus TCP/IP slaves. Data from them can, if required, also be imported into the IMx but this is entirely optional and unconnected to their use as a Direct Modbus device.
- For clarity, be aware that there is also a third variant of Modbus, usually called Modbus over TCP/IP, where Modbus RTU messages are sent over TCP/IP. This variant is not supported by IMx devices.

# 4 Modbus and IMx

## 4.1 The Modbus Protocol

The protocol supports inter device communication where each device must be configured as either a master or a slave.

On RS-485 infrastructure, this is strictly, one master and one or more slave devices per bus. On Ethernet infrastructure multiple Modbus TCP/IP instances can be supported simultaneously so, (subject to some caveats), an IMx device can be both a Modbus TCP/IP master and a slave device.

Again, subject to some restrictions, Modbus RTU and Modbus TCP/IP functionality can be used simultaneously, on the same IMx device. A summary is provided in the table below, with explanations in the text that follows:

*Table 3 Allowable Modbus communication combinations*

| | **TCP Master** | **RTU Master** |
|---|---|---|
| **TCP Slave*** | Supported | Supported |
| **RTU Slave** | Supported | Not supported |
| **TCP Slave* + RTU Slave** | Supported | Not supported |

Notes:

*The IMx can be a slave in multiple TCP connections subject to the following limitation:

- A total, maximum of 4 Modbus connections (master, slave, TCP and RTU, including where the IMx is a TCP master to a Direct Modbus device)
    - If the IMx is a Modbus RTU slave on a bus with other slaves, this only counts as one connection towards the above limitation as only one master connects to it.

Refer also to the implementation notes and descriptions, below.

**IMx Modbus Slave**

- Modbus slaves respond to requests from a Modbus master.
- A Modbus slave can export all its measured analogue vibration, process data, derived and digital speed points to the Modbus input registers 1 to 128 at the request of any kind of Modbus master.
- A Modbus slave can import values to the IMx holding registers 1 to 32, also known as Modbus external communication channels, at the request from any kind of Modbus master. See also Notes on Implementation.

- Slave mode accepts the following operations:
  - Read input registers (function code 4).
  - Write multiple registers (function code 16). Writing to holding register 1-32 writes to Modbus external channels 1-32.
  - Read holding registers (function code 3) to read back holding registers 1-32. Initial value -32768 = Cable fault (CF).

**IMx Modbus Master**

- An IMx Modbus master can read data from up to four, Modbus slave devices but noting that:
  - The number of slaves supported is reduced by the number of other Modbus connections, enabled.
  - When using the Modbus RTU link to gather GPS data from the external receiver, CMON 4139, @ptitude Observer provides a specific External communication Type, GPS (GPS50M) that appropriately configures a Modbus RTU Master connection and the Modbus external channels for the GPS data.
- When an IMx master is appropriately configured, the IMx holding registers 1 to 32 can be used as external channels to read data from a slave (or slaves). Modbus external channels are available for analysis and gating in the same way as local measured channels.
- Where the 32 holding registers are shared across more than one slave device, the implementation supports each slave having a different byte/word order (endianness/swap parameters).
- In master mode, the IMx polls the assigned slave (or slaves) as configured. In @ptitude Observer, Modbus groups are used to optimise the data transfer and avoid unreadable registers. Up to three groups per slave, each with a maximum span of 125 registers are supported. Make sure to select a different Modbus group if this register range is exceeded (example 3 below), if there are unreadable registers to avoid or if the register type changes (example 4 below):

  1] Register type: Holding, Register number: 101, Group 1

  2] Register type: Holding, Register number: 201, Group 1

  3] Register type: Holding, Register number: **409**, Group **2**

  4] Register type: **Input**, Register number: 150, Group **3**

**Notes on Implementation**

- An IMx Modbus master only polls for data, to populate Modbus external channels.
  - On one IMx, only one Modbus instance can be a Modbus master.
  - Master polling rate is about 1 to 2 times per second.
  - @ptitude Observer initiated polling of a Direct Modbus device is every minute or slower (configurable).

- When used as a slave, data can usually be both imported and exported but noting that:
    - Export data only, if another Modbus instance is a Modbus master i.e. the 32 Modbus external channels cannot be split between a master and a slave.
    - It is not allowed to write to IMx registers when the IMx is a slave to multiple masters.
- Mapping of exported measurement points to Modbus input registers is as shown in Table 4, below.
- Mapping of imported Modbus holding or input registers to Modbus external channels is as shown in Table 5, below.
- The read function codes are 0x03 and 0x04.
- The write function code is 0x10.
- When the IMx is a Modbus master, the data type can be:
    - 16-bit signed integer or Int16*
    - 32-bit signed integer or Int32*
    - 16-bit unsigned integer or UInt16*
    - 32-bit unsigned integer or UInt32*
    - 32-bit floating point or Float*
- When the IMx is a Modbus slave and imports data, the data type is:
    - 16-bit signed integer or Int16*
- When the IMx is a Modbus slave and exports data, the data type can be:
    - 16-bit signed integer or Integer 16 bit*
    - 32-bit floating point or Float 32 bit*

* Notation/term used in @ptitude Observer

- Modbus slave speed channels are transmitted in cycles per minute (cpm).
- Whilst the number of physical relay driver outputs is IMx type dependent (e.g. two on the IMx-8/IMx-16 family and four on an IMx-S 16), all IMx transfer 16 ‘relay’ states in the Relay mask. The remainder are virtual Modbus outputs.

Register Terminology

- 3xxxx = Input registers (read-only), first register is 30001
- 4xxxx = Holding registers (read/write), first register is 40001
- The offsets above are automatically applied based on the register type chosen. In the @ptitude Observer user interface, the first registers of each have register numbers of either ‘0’ or ‘1’ depending upon the context:
    - When referring to IMx device registers, register 1 is the first register.
    - When referring to the registers of an external device or system, the first register is register 0.

## 4.2 Modbus Register, Modbus Channel and Measured Channel

The correspondence between exported measurement points and Modbus input registers is listed in the following table.

*Table 4 Correspondence Between Exported Measurement Points and Modbus Input Registers*

| Modbus Input Register | Integer output | Float output |
|---|---|---|
| 30001 to 30016 | Configurable measurement point* | |
| 30017 | Speed digital channel 1 | Speed digital channel 1 |
| 30018 | Speed digital channel 2 | |
| 30019 | Speed digital channel 3 | Speed digital channel 2 |
| 30020 | Speed digital channel 4 | |
| 30021 | Speed digital channel 5 | Speed digital channel 3 |
| 30022 | Speed digital channel 6 | |
| 30023 | Speed digital channel 7 | Speed digital channel 4 |
| 30024 | Speed digital channel 8 | |
| 30025 | Relay mask ** | |
| 30026 | Warning mask** | |
| 30027 | Alarm mask** | |
| 30028 | Heartbeat*** | |
| 30029 | System information and digital input state<br>Bit 0-7: Digital input channel 1-8, set at 0 or 1 depending on the state of the respective digital channel input.<br>Bit 8-13: Reserved<br>Bit 14: When set at 1, the IMx is connected to the @ptitude Observer Monitor server.<br>Bit 15: When set at 1, the IMx is in maintenance mode. In this mode no measurement data is stored, refer to the relevant IMx user manual for further information. | |
| 30030 | System diagnostic status<br>Bit 0-1: 0=No alarm, 1=Warning, 2=Alarm, 3=Reserved<br>Bit 2-15: Reserved | |
| 30031 to 30128 | Configurable measurement point * | |

Notes to table 4:

*Any IMx measurement point can be selected and exported via Modbus on these registers except Counter, Running hours and Gear inspector points.

Exported vibration point data can relate not only to the overall value but may comprise multiple registers to export amplitudes for the bands/components F1 to F4, as well.

**Mask is a 16-bit value which shows the condition of all channels or state of all relays in a hexadecimal format. See Bit Representation of Mask Table 6, for detailed information. Note that for the Relay mask, the full 16-bits are potentially used as in addition to whatever physical relay driver outputs the particular IMx device has, the balance of 16 are assignable in @ptitude Observer as virtual, Modbus outputs.

***Heartbeat register value (one-indexed) will update every read with non-zero positive values. To use it, verify that there is a difference between the last and next value.

The correspondence between imported Modbus input and holding registers and Modbus external communication channels is as shown in the following table.

*Table 5 Correspondence Between Imported Modbus Input and Holding Registers, and Modbus External Communication Channels*

| Modbus external Channel | Imported Modbus Input Register when IMx is a Modbus Master | Imported Modbus Holding Register when IMx is a Modbus Slave |
|---|---|---|
| 1 | Configurable | 40001 |
| to | | |
| 32 | Configurable | 40032 |

## 4.3 Bit Representation of Mask

One bit is allocated per channel, each hex digit of the Relay, Warning and Alarm Masks therefore conveys information for 4-channels. The following table illustrates how all hex (single) digit values correspond to channel/relay numbers.

*Table 6 Hex Representation for Relay, Warning and Alarm Masks*

| Hexadecimal Representation | Channel/Relay Numbers |
|---|---|
| 0x0001 | 1 |
| 0x0002 | 2 |
| 0x0003 | 2, 1 |
| 0x0004 | 3 |
| 0x0005 | 3, 1 |
| 0x0006 | 3, 2 |
| 0x0007 | 3, 2, 1 |
| 0x0008 | 4 |
| 0x0009 | 4, 1 |
| 0x000A | 4, 2 |
| 0x000B | 4, 2, 1 |
| 0x000C | 4, 3 |
| 0x000D | 4, 3, 1 |
| 0x000E | 4, 3, 2 |
| 0x000F | 4, 3, 2, 1 |
| Then similarly: | |
| 0x0010 to 0x00F0 | 5 to 8, 7, 6, 5 |
| 0x0100 to 0x0F00 | 9 to 12, 11, 10, 9 |
| 0x1000 to 0xF000 | 13 to 16, 15, 14, 13 |

As an example, if all channels were in warning other than Channel 11, Register 30026 would read 0xFBFF (binary 1111 1011 1111 1111), if only Channel 11 were in warning, register 30026 would read 0x0400 (binary 0000 0100 0000 0000).

For the Relay mask, the 16-bits reflect the state of the 16, IMx relay outputs (the physical relay driver outputs and the remainder being Modbus, only, outputs). For relays a ‘1’ means energised, in warning or alarm state.

# 5 Configuring Modbus Capability for IMx

## 5.1 Configuration Scenarios

IMx devices are normally configured by the SKF @ptitude Observer software. Refer to the SKF @ptitude Observer User Manual.

There are different procedures for configuring Modbus capabilities depending on the scenario. For an overview of each, select a matching scenario and follow the steps.

- Export Data from IMx Modbus Slave to Generic Modbus Master
- Import Data from Generic Modbus Master to IMx Modbus Slave
- Import Data from Generic Modbus Slave to IMx Modbus Master
- Data Transfer between IMx devices, refer Appendix A

## 5.2 Export Data from IMx Modbus Slave to Generic Modbus Master

IMx Modbus slaves can export data to a generic Modbus master via Modbus input registers 1 to 128:

- All analogue measurement point types of IMx devices, except Gear inspector points.
- All digital measurement point types of IMx devices, except Counter and Running hours points.

To configure the IMx, Modbus slave, follow these steps:

1. Have measurement points setup for the data being transferred
2. Configure as a Modbus slave and allocate it a unique **slave address**
3. Select the desired **Value type** (data type) for the Modbus data transfer
4. Then using the **Data Export** functionality:
   - Pick a register and set the specific measurement point data it will hold
   - **Register** will be in the range 1 to 128 (to 127 if floating point)
   - As an input register, Table 4, this will correspond to 30000 + **Register**

Notes

- ➢ In case of problems establishing the Modbus communication, check that the communication parameters agree: Modbus RTU, (**Bps**, **Parity** and **Stop bits**), Modbus TCP/IP (slave **IP address** and **Port** number) and refer to Interface and Troubleshooting.
- ➢ Further guidance on configuration aspects can be found in the SKF @ptitude Observer User Manual.

## 5.3 Import Data from Generic Modbus Master to IMx Modbus Slave

IMx Modbus slaves can usually import values from a generic Modbus master via Modbus holding registers 1 to 32 but refer Notes on Implementation for some restrictions applying to this capability.

1. For the IMx device which is to import data:
   - Configure as a Modbus slave
   - Configure appropriately the slave address and relevant communication parameters
2. Then using the **Data import** functionality:
   - Select which of the 32 holding registers will be used, noting that in this configuration there is a fixed '1:1' relationship between external channel and holding register number, refer Table 5.
   - For each:
     - Make it **Active**
     - Give it a **Name** and an **E.U.**
     - Change the default **Zero level** and **Full Scale [E.U]** as required
     - Note that these are the only settings that are configurable
3. Now in the hierarchy, add an IMx Process point for each external channel:
   - In **Device**, choose the IMx acting as the slave (receiving the data)
   - In **Channel**, the Modbus external channels will appear as 101 to 132

Notes

- In case of problems establishing the Modbus communication, check that the communication parameters agree: Modbus RTU, (**Bps**, **Parity** and **Stop bits**), Modbus TCP/IP (slave **IP address** and **Port** number) and refer to Interface and Troubleshooting.
- Further guidance on configuration aspects can be found in the SKF @ptitude Observer User Manual.

## 5.4 Import Data from Generic Modbus Slave to IMx Modbus Master

An IMx Modbus master can import data from generic Modbus slaves, regardless of whether they are holding or input registers, to Modbus external communication channels, 1 to 32.

1. For the IMx device which is to import data:
   - Configure as a Modbus master
   - Add a slave at the applicable address
2. Then using the **Data import** functionality:

- Associate the slave's Modbus register with a Modbus external channel
- Modbus external communication channel will be in the range 1 to 32
- For the selected channel:
    - Make it **Active**
    - Pick the applicable **slave name**
    - Choose **Register type** and enter the **Register Nr** (0-base)
    - Select the **Data type** (value type) that the slave is making available
    - Give it a **Name** and an **E.U.**
    - Change any of the other default settings, as required

3. Now in the hierarchy, add an IMx Process point, for each external channel:
    - In **Device**, choose the IMx acting as the master (receiving the data)
    - In **Channel**, the Modbus external channels will appear as 101 to 132

Notes

- In case of problems establishing the Modbus communication, check that the communication parameters agree: Modbus RTU, (**Bps**, **Parity** and **Stop bits**), Modbus TCP/IP (slave **IP address** and **Port** number) and refer to Interface and Troubleshooting.
- Further guidance on configuration aspects can be found in the SKF @ptitude Observer User Manual.

# 6 IMx device in stand-alone mode

When an IMx from the IMx-8, IMx-16 family of devices is configured as stand-alone with machine templates, all the Modbus configuration is set via SKF @ptitude Observer and the only parameters editable within IMx Manager are the communication parameters, meaning that the Modbus communication can be enabled or disabled or changed from RTU to TCP/IP. The different registers configured are not editable from IMx Manager.

When it is configured as stand-alone with manual configuration, the Modbus communication parameters are configured by default as:

- Modbus TCP/IP
- Modbus slave
- Port 502

Note that the Modbus communication parameters are also editable when Modbus RTU is selected.

For a Modbus slave device, all registers are automatically configured with respect to the different measurement points created. These registers are configured in floating point format so no scale is needed. When using floating point, every measurement will use a pair of registers (i.e. measurement 1 on registers 1 and 2, measurement 2 on registers 3 and 4).

*Table 7 Modbus Slave Device Automatic Register Configuration*

| Point Type | DBID | Modbus Register | Trend/Modbus Unit |
|---|---|---|---|
| Digital | 99990000 + ch | - | - |
| Speed | 99991000 + ch | 17 + 2(ch – 1) | cpm |
| Process | 99992000 + ch | 1 + 2(ch – 1) | °C |
| Vibration (acceleration) | 99995000 + ch | 33 + 2(ch – 1) | g pk |
| Vibration (velocity) | 99996000 + ch | 49 + 2(ch – 1) | mm/s rms |
| Vibration (envelope) | 99997000 + ch | 65 + 2(ch – 1) | gE p-p |

ch = channel number

# 7 Interface and Troubleshooting

## 7.1 Procedure

To check the Modbus communications, perform the following steps:

1. Start @ptitude Observer Online Device Configurator program located in the @ptitude Observer program folder.

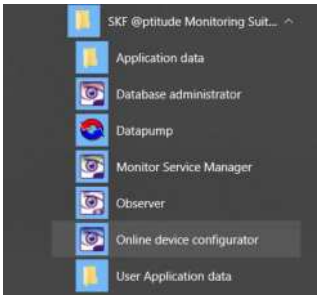



*Figure 7 Online Device Configurator shortcut*

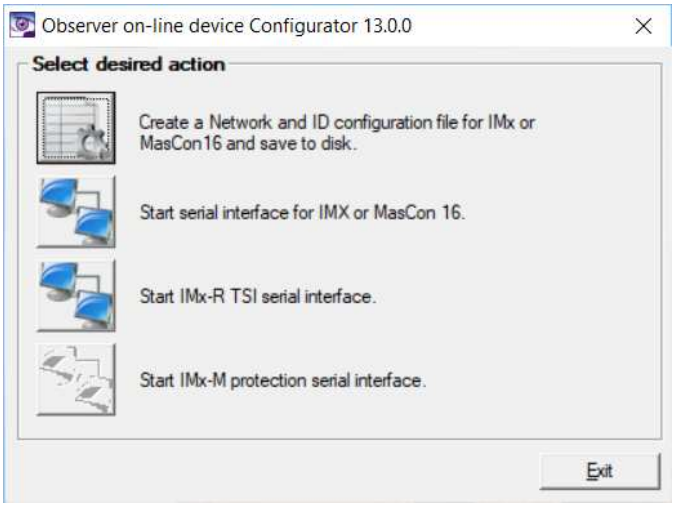



*Figure 8 On-line device Configurator, Desired Actions*

2. On the opening window, click Start serial interface for IMx. The Serial interface dialog opens.

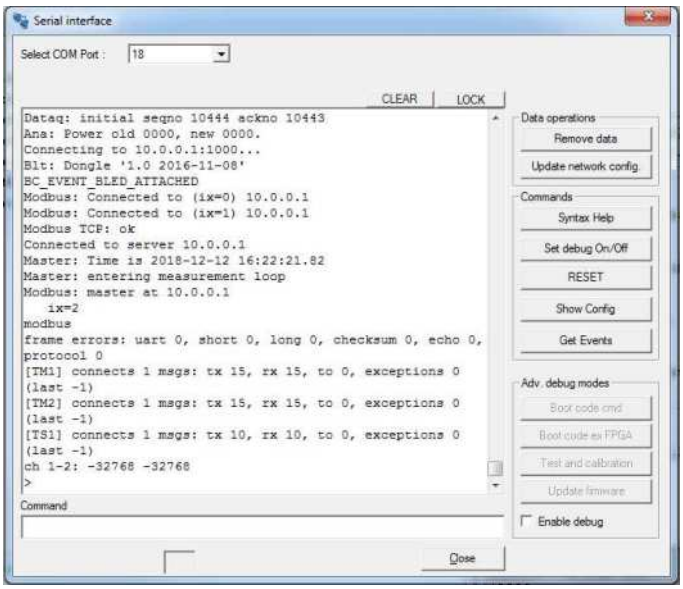



*Figure 9 On line device Configurator, Serial Interface Dialog*

3. On the Serial interface dialog, select the COM port number and enter Modbus in the Command box.
    - Statistics on communication and the contents of the registers will appear on the screen. The statistics include:
        - Frame errors (short and long) and Checksum errors
    - Then the following statistics are included for each Modbus connection:
        - The number of messages sent (tx) and received (rx)
        - The number of request timeouts (to) and exceptions
    - The Modbus connection identifier, example (TM1), has the following key:
    - T/R - TCP or RTU
    - M/S - Master or slave
    - number - slave id
4. A properly working Modbus communication link should exhibit increasing numbers of messages, sent and received, but it should not exhibit significant increases in error or timeout counts.
5. In case of errors or timeouts, check the following:
    - That the RS-485 cable is correctly physically connected as described in Modbus over RS-485.
    - The transmission characteristics are defined correctly as described in Transmission Characteristics.
    - The Modbus master-slave pair address is entered correctly as described in Configuring Modbus Capability for IMx.

- Verify that each IMx device, configured as a Modbus slave in an RS-485 bus, has a unique slave number.

6. This process of checking the Modbus communication can be repeated, as required during the installation/test, to diagnose the communication activity or the lack of it.

# Appendix A - Data Transfer between IMx devices

Transfer of measurement data from one IMx device to another IMx device is normally used for test only. To transfer data between two IMx devices, follow these steps.

1. Identify which device is to export and which device is to import data
2. The device which is to export data:
   - Have measurement points setup for the data being transferred
   - Configure as a Modbus slave and allocate it a unique slave address
   - Select **Value type** (data type) for the Modbus data transfer
3. Then using the **Data Export** functionality:
   - Pick a register and set the specific measurement point data it will hold
   - Register will be in the range 1 to 128 (to 127 if floating point)
   - As an input register, Table 4, this will correspond to 30000 + Register
4. For the other IMx device which is to import data:
   - Configure as a Modbus master
   - Add a slave at the address allocated in step 2
5. Then using the **Data import** functionality:
   - Associate the slave's input register with a Modbus external channel
   - Modbus external communication channel will be in the range 1 to 32
   - For the selected channel:
     - Pick the applicable **slave name**
     - Choose Input as **Register type** and enter the **Register Nr** (0-base)
     - Select the **Data type** (value type) that the slave is making available
6. Now in the hierarchy, add an IMx Process point, for each external channel:
   - In **Device**, choose the IMx acting as the master (receiving the data)
   - In **Channel**, the Modbus external channels will appear as 101 to 132

Notes

- In case of problems establishing the Modbus communication, check that the communication parameters agree: Modbus RTU, (**Bps**, **Parity** and **Stop bits**), Modbus TCP/IP (slave **IP address** and **Port** number) and refer to Interface and Troubleshooting.
- Further guidance on configuration aspects can be found in the SKF @ptitude Observer User Manual.

# Appendix B - Older and legacy devices

## B-1 Overview

The design of the current IMx family (IMx-8, IMx-8Plus, IMx-16 etc.) is somewhat different from the older/legacy SKF Multilog IMx-S, IMx-W, IMx-C, IMx-B and IMx-T devices. This appendix includes further information of note when working with those older/legacy devices.

## B-2 Related documentation

*Table 8 Related documentation*

| Document | Part No. |
|---|---|
| IMx-S User Manual, Rev S or later | 32087700 |
| IMx-W User Manual, Rev T or later | 32146100 |
| IMx-C User Manual, Rev E or later | 32264900 |
| IMx-B User Manual, Rev D or later | 32293200 |
| IMx-T User Manual, Rev G or later | 32096300 |

## B-3 Important messages

These devices tend to have more functionality manually configured by board mounted DIP switches. Due to this, always observe the following important precautions:

*IMx devices contain circuit boards that are static sensitive. Therefore, use appropriate precautions to prevent Electrostatic Discharge (ESD) when handling circuit boards.*

*DIP switch settings must be handled with special care to prevent any damage to the IMx device:*

*Do NOT change DIP switch settings while the IMx device is powered, as this may cause damage and void the warranty.*

*Before powering up the IMx device, make sure that DIP switch settings are properly set to match the recommendations for the connected sensors. Incorrect settings may cause permanent damage to the IMx device.*

## B-4 Terminal blocks

Some examples from a selection of legacy IMx devices are shown in the figures below:


a) IMx-S b) IMx-W

*Figure 10 RS-485 connection examples*

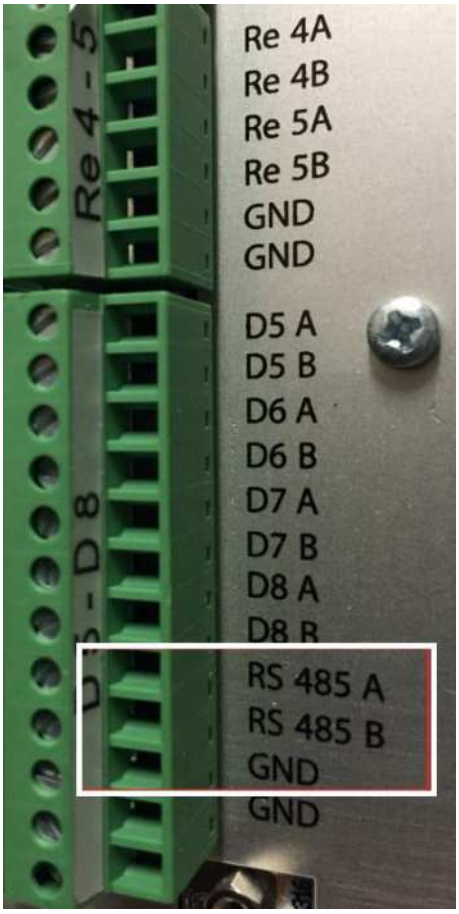



*Figure 11 IMx-T RS-485 terminals*

## B-5 RS-485 Bus Termination

SKF Multilog IMx-S, IMx-W, IMx-C, IMx-B and IMx-T devices have internal termination which is manually controlled by a DIP switch.

- Note that for all DIP switches a setting of '1' corresponds to ON and a setting of '0' corresponds to OFF.

If an IMx device is at the end or beginning of the RS-485 Bus, activate the built-in termination resistor by setting the DIP switch.

### B-5-1 IMx-S, IMx-C and IMx-B

DIP21 for IMx-S, IMx-C and IMx-B should be set as described in the table below. To access DIP21 on the IMx-S, remove the front panel. On newer systems, there is a hole in the front panel for DIP21 (no need to remove the front panel).

*Table 9 RS-485 Termination DIP21 settings – board version lower than v1.24*

| DIP21 settings 1234 | Functionality with I/O board version less than v1.24 (DIP21-switch with 4 positions) |
|---|---|
| 0100 | RS-485 Termination resistor enabled |
| 0000 | RS-485 Termination resistor disabled |

*Table 10 RS-485 Termination DIP21 settings - board version v1.24 or higher*

| DIP21 settings 123456 | Functionality with I/O board version v1.24 or higher (DIP21-switch with 6 positions) |
|---|---|
| 011010 | RS-485 Termination resistor enabled |
| 001010 | RS-485 Termination resistor disabled |

IMx-S, IMx-C and IMx-B Modbus termination DIP switch (DIP21) boards pre v1.24:

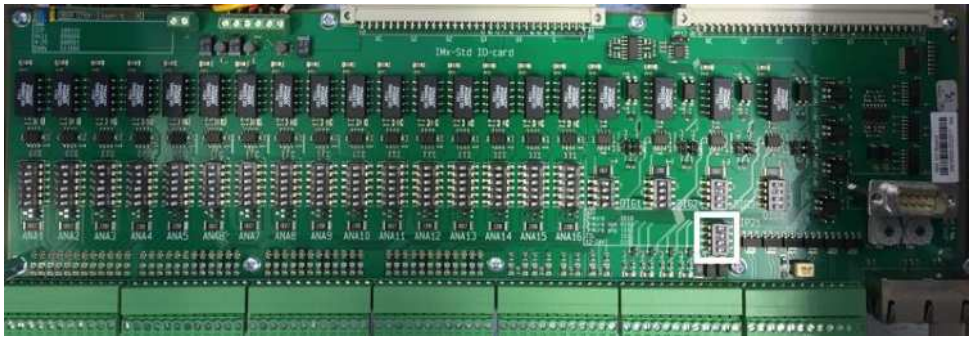


*Figure 12 Modbus Termination DIP Switch, Example for DIP21 (pre v1.24 board)*

### B-5-2 IMx-W

Set DIP3 for IMx-W as in the table below.

*Table 11 RS-485 Termination DIP3 Settings*

| DIP3 settings 1234 | Function |
|---|---|
| 1000 | RS-485 Termination resistor enabled (default) |
| 0000 | RS-485 Termination resistor disabled |

### B-5-3 IMx-T

Set ANA21 for IMx-T switch number 2 to ON. ANA21 can be accessed after removing the IO board from the IMx-T rack.

*Table 12 RS-485 Termination ANA21 Settings*

| ANA21 settings 1234 | Function |
|---|---|
| 0100 | RS-485 Termination resistor enabled |
| 0000 | RS-485 Termination resistor disabled |

IMx-T Modbus termination DIP switch (ANA21):

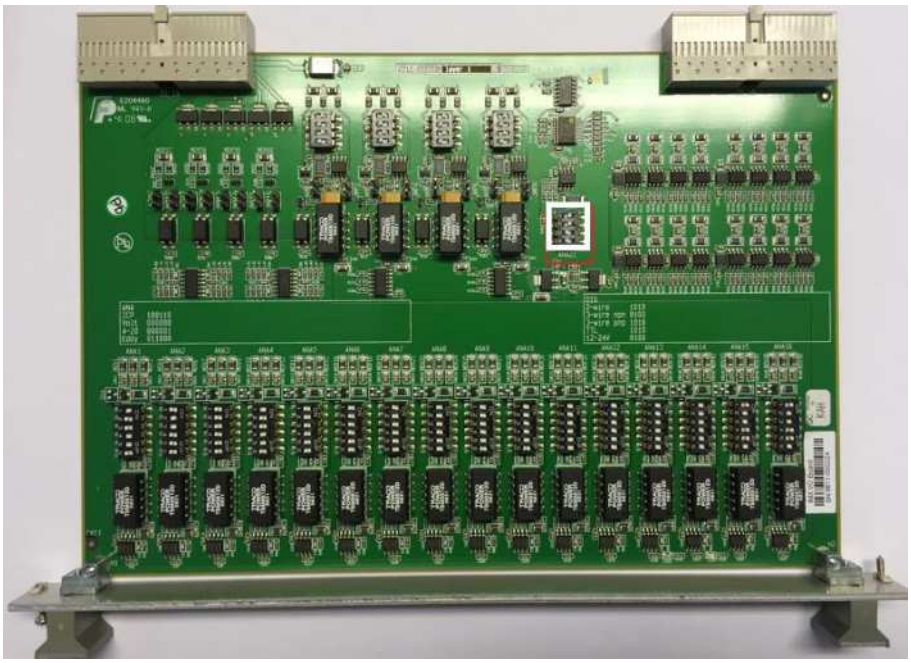


*Figure 13 Modbus Termination DIP Switch, Example for ANA21*