smartPREDICT-AE RS485/USB

Digital vibration and motion sensor system for IIoT

iNDTact GmbH
Friedrich-Bergius-Ring 15
97076 Würzburg
Germany
Tel.: +49 (0) 931 2999 7 330
Fax.: +49 (0) 931 2999 7 325
Email: info@indtact.de
Website: www.indtact.de

Modbus Reference Guide

1 Table of Content

2 Revision History

Revision	Date	Author(s)	Description
V20240202	02.02.2024	JW, YK	Add description for new MEMS ADXL373; Add description about order analysis; Add GPIO mode “Pulse counter per second”; Add Mea Reg “Pulses per second” and “RPM rounds per minute”; Add Mea Reg “PSD df” and “PSD dn”
V20231114	14.11.2023	JW	Add description for new MEMS LSM6DSO32X; Add description about alternative main sensor ADXL1001
V20221027	27.10.2022	JW	Add sPRs, sPRc, SPRsc parameters; Add Description about GPIO Channel (chapter 5.2 and 6.4.3); Add Sensor Mode “Continuous GPIO Triggered Start / Stop”; Add Measurement Registers “Elapsed Window Length”
V20211129	29.11.2021	JW	Fix in description of PSD in Data Recorder (chapter 6.4.7)
V20210713	13.07.2021	JW	Fix in description of lower/upper thresholds of Autonomous Condition Monitoring; Add Measurement register addresses for Electrical Conditions
V20210511	11.05.2021	JW	Remove frequency min setting from chapter “AE sensor analog signal filter”; Update Product Description; Update Format of Revision
V20210304	04.03.2021	JW	Initial release

3 General

This application note describes how to communicate with a smartPREDICT device via Modbus RTU. Please refer to the user guide for wiring options and specifications.

Modbus is a serial communications protocol for industrial electronic devices. It enables communication among many devices connected to the same network. The basic architecture is based on a single master device and up to 247 slave devices. Many hardware interfaces and software libraries provide APIs for Modbus communication.

This guide describes the Modbus RTU interface which is based on a serial communication interface.

3.1 Product Description

The smartPREDICT-AE is a multi-sensor system based on iNDTact's high-bandwidth vibration sensor iMPactXS that enables highly sensitive vibration-based condition monitoring of structures, products and processes. Alongside the high-bandwidth, high-sensitivity vibration sensor, a micro-electromechanical system (MEMS) offers additional motion information (acceleration and angular velocity). The temperature sensor on the bottom of smartPREDICT-AE allows for internal reference temperature measurement next to the iMPactXS.

The smartPREDICT-AE can be used for direct parameter measurement and transmission to machines (PLCs) or edge devices for I-IoT via Modbus RTU.

Key features:

- Integrated ultra-low noise acoustic emission sensor with 96 kHz sampling rate
- Integrated acceleration sensor
- Integrated gyroscope sensor
- Integrated temperature sensor
- Autonomous condition monitor with teach-able fingerprinting and alarm output
- Output of 11 vibration features (sPR, RMS, etc.) for each Sensors (AE/Acc/Gyro)
- Configurable digital IIR filters for band-width and POI control
- Real time raw data streaming
- Recorder for Raw Data and Spectrum
- GPIOs for digital/analog input or output
- Serial Modbus RTU interface with switchable line termination
- USB interface*
- Firmware updates via bootloader for new features or custom analysis
- IP67 protection

Applications:

- Conditional monitoring of machines & processes with direct input for PLC
- Monitoring and trending of vibrational parameters of high value assets
- Shock and vibration monitoring for con-struction machine/vehicle or agricultural machine/vehicle
- Materials research
- Production control
- Final inspection and testing
- Incoming goods department
- Biological studies
- Structure monitoring

4 Modbus RTU

Modbus RTU is serial bus protocol running on a three-wire interconnection. The default serial port settings are shown below under “Connection”.

4.1 Introduction

All messages on a Modbus serial bus are based on the following telegram structure. The master device sends requests to slaves, which are addressed by the value of the first field. All telegrams are secured by a CRC16 checksum.

Depending on the function code more parameters and content is added to a telegram. A list of supported function codes is listed in section “Supported Function Codes”. Tables of supported registers are listed in section “Supported Registers”.

General message form

Device Address	Function Code	Data	Checksum
8 bits	8 bits	n * 8 bits	16 bits

Please note, all register addresses are transmitted relative to the base-address set by the function code.

Example: Read Acoustic Emission Sensor RMS value

The process of master / slave communication is shown in the following example. Here a master requests the values of two 16-bit registers at register address 0x31008. The relative register address is 0x1008.

Request:

Dev Addr	FC	Data		Checksum
		Start Addr	No. of registers	
0x01	0x04	0x1008	0x0002	0xF4C9

Device 0x01, 0x04 Read Input register, Read 2 registers starting at addr. 0x1008 (0x31008)

Answer:

Dev Addr	FC	Data		Checksum
		No. of data (16-bit)	Information	
0x01	0x04	0x02	0x3DCC 0xCCCD	0x2A82

Device 0x01, 0x04 Read Input register, 2 * 2 Byte appended, value 0.1 (0x3DCCCCCD)

4.2 Connection

The default address of this device is 1 and can be set to 1…247. Please ensure, that there are not two devices with the same address on one RS485 bus. In such a case, an abnormal behavior of the whole bus can occur.

Setting	Default Value
Baud	19200
Data Bits	8
Stop Bits	1
Parity	Even
Hardware Flow Control (RTS)	Disable
Software Flow Control (XON/XOFF)	Disable

This default mode corresponds to “8E1” in data/parity/stop (D/P/S) conventional notation.

Visual Diagnosis

The STATE LED near the connector terminal displays active communication. The USER LED is either controlled by Modbus commands or by alert thresholds.

Color of STATE LED	Description
Green	Bus message not addressed to the device
Yellow	Bus message addressed to the device
Red	Frame reception or parity fault

The STATE LED is controlled internally by an one-shot-timer and will be switched on for approximately 1 millisecond for each received message.

4.3 Communication Settings

4.3.1 Change device (slave) address

To allow multiple smartPREDICT devices to be connected at the same bus connection, each device must be given an unique address in the range 1…247.

Example: Update device #1 to the new device address #9.

Dev Addr	FC	Data		Checksum
		Start Addr	No. of registers	
0x01	0x06	0x0011	0x0009	0x19C9

Device 0x01, 0x06 Write Single register, Write to register address 0x0011 (0x4011), New slave address is 9

Example: Change address of a connected but unknown device

This request uses the broadcast address 0 to reach all connected devices. In this specific example, please ensure only one smartPREDICT is connected.

Dev Addr	FC	Data		Checksum
		Start Addr	No. of registers	
0x00	0x06	0x0011	0x0009	0x1818

Broadcast, 0x01, 0x06 Write Single register, Write to register address 0x0011 (0x4011), New slave address is 9

Note: Changing Device address is only valid during current runtime. To make setting persistent, write coil “Save Configuration” afterwards.

4.3.2 Change baud rate

This device supports multiple transfer speeds for Modbus communication. Please check the Modbus master if it supports the baud rate you consider using.

All supported baud rates listed below. Only common baud rates can be saved persistent, un-common baud rates are only valid for current runtime.

Common baud rates: 4800, 9600, 19200 (default), 38400, 56000, 57600, 115200

Un-common baud rates: 128000, 230400, 256000, 460800, 500000, 576000, 921000, 1000000, 1500000, 2000000, 3000000, 6000000, 12000000, 13500000

(For Hardware Revision 4 (MC4 0 x), only baud rates up to 500000 can be configured)

Example: Set baud rate to 9600

Dev Addr	FC	Data		Checksum
		Start Addr	No. of registers	
0x01	0x06	0x0013	0x0003	0x380E

Device 0x01, 0x06 Write Single register, Write to register address 0x0013 (0x40013), New baud rate is 9600

4.4 Supported Function Codes

Following Modbus Function Codes (FC) are supported:

FC	Name	Reg.	Usage
0x01	Read Coils	0x0xxxx	Reading device status
0x03	Read Holding Registers	0x4xxxx	Reading device configuration
0x04	Read Input Registers	0x3xxxx	Reading measurements
0x05	Write Single Coil	0x0xxxx	Setting device runtime flags
0x06	Write Single Register	0x4xxxx	Setting device configuration
0x08	Diagnostic	-	Testing device connection
0x10	Write Multiple Registers	0x4xxxx	Setting device configuration
0x15	Write File Record	-	Firmware Update only
0x2B	Read Device Identification	-	Read Device Identification

Sub-Function Codes for Diagnostic

Sub-FC	Name	Notes
0x00	Return Query Data	Check device availability
0x01	Restart Communications Option	Reboot Device, see [SFC1]
0x0A	Clear Counters	Clear diagnostic counters
0x0B	Return Bus Message Count	
0x0C	Return Bus Communication Error Count	
0x0D	Return Slave Exception Error Count	
0x0E	Return Slave Message Count	
0x0F	Return Slave No Response Count	
0x10	Return Slave NAK Count	
0x11	Return Slave Busy Count	
0x12	Return Bus Character Overrun Count	

[SFC1] Reboot Device
The reboot is performed immediately. A response will not be sent

Sub-Function Codes for Read Device Identification

Sub-FC	Name	Notes
0x00	Vendor Name	
0x01	Product Code	
0x02	Firmware Revision	
0x03	Vendor URL	
0x04	Product Name	
0x05	Model Name	
0x06	Application Name (reserved)	

4.5 Data Encoding

Standard Modbus data types are Byte (8-bit) and registers (16-bit). The default byte order in telegrams is big-endian. All extended data types (32-bit values) are clarified in this section. Signed integers are stored using two's complement representation.

Type	Short	Reg. Len.	Bytes	Sign	Description
bool	bool	1	0	U	Single bit
int16	i16	1	2	S	Signed 16-bit integer -32768 - 32767
uint16	u16	1	2	U	Unsigned 16-bit integer 0 - 65535
uint32	u32	2	4	U	Unsigned 32-bit integer 0 - 4294967295
float32	f32	2	4	S	Signed 32-bit floating point number

Unsigned 32-bit integers
Values in uint32 format are stored in two 16-bit registers. The byte order is big-endian. The MSB (high) byte and MSB (high) word are transmitted first.
Example: 4000123456 (0xEE6D0A40) is transmitted as 0xEE 0x6D (MSB (first) register), 0x0A 0x40 (LSB (second) register).

Floating point values
Transmitting floating point values over Modbus requires access to two registers. Floats are stored in big-endian byte order according to IEEE-754.
Example: 123.125f (0x42F64000) is transmitted as 0x42 0xF6 (MSB (first) register), 0x40 0x00 (LSB (second) register).

	Sign	Exponent	Mantissa
Value	+1	133	7749632
Encoded	0	10000101	11101100100000000000000

5 Function Units

The device provides multiple function units to control the device and read the measurements.

5.1 Device Management and Communication

This module consists of readable information about the device and the status. The Modbus RTU communication settings are organized here as well.

5.1.1 Device Management

The device management only uses configuration registers and status coil registers and provides the following information and configuration options.

The configurable options for the “USER LED Control” and “Float32 write test” are only valid during current runtime. Writing coil “Save Configuration” won’t make these settings persistent.

Device Ready
The device has a “Device Ready” flag in the status coil registers. This will be set to ‘1’ if device finished booting and is ready for operation.

Device Selftest
At every system start, the device performs a (startup-) selftest, where all hardware units are tested. The result of the selftest can be read in the status coil register “Selftest passed”, which is set to ‘1’ if selftest is passed.

A detailed selftest result can be read in the configuration register “Selftest Result”. The result is binary coded into the “Selftest Result” value. A ‘1’ is a passed test, a

- Systemclock selftest
- LED’s selftest
- Accelerometer / gyroscope sensor selftest
- External temperature sensor selftest
- chAMP unit selftest
- SDRAM selftest
- External flash selftest
- Analog values selftest
- AE sensor selftest
- Modbus CRC unit selftest
- EEPROM selftest

A failed selftest is also displayed via the USER LED, which toggles 10 times red/off after powering up the device. Then the USER LED will flash once for each performed selftest item.

- Green: Selftest passed
- Red: Selftest failed

Unique Hardware ID
The device has a hardware ID which is unique for every device. It consists of 8 characters with the numbers 0…9 and letters A…F. The “Unique Hardware ID” configuration register is a 32-bit value and the resulting number must be interpreted as a HEX value which represents the Unique Hardware ID.

Example: “Unique Hardware ID” as read from the configuration register is HEX-value 0xA3001E19 and results in the unique hardware ID A3001E19

Hardware Version
The hardware version of the device can be read in the configuration register “Hardware Version” and is equal to the read uint16 value.

Firmware Version
The firmware version can be read from the configuration register “Firmware Version”. The version is a 32-bit value which stores the MAJOR-, MINOR- and PATCH-Version + a reserved value.
For each part, the size of a byte is used and they are stored the following way:
MAJOR | MINOR | PATCH | Reserved

Example: “Firmware Version” as read from the configuration register is HEX-value 0x021F0100 and results in the firmware version 2.31.1

USER LED Control
The user can control the USER LED with Modbus commands. The following colors can be set:

- off
- blue
- green (default)
- cyan
- red
- magenta
- yellow
- white

Some function units are able to change the color, like the autonomous condition monitoring. This will overwrite the user setting.

Float32 write test
A float32 write test register is provided for loopback write- and read tests of the float32 data type.

5.1.2 Communication

The communication settings only uses configuration registers and provides the following configuration options.

All settings are only valid during current runtime. Writing coil “Save Configuration” to make this settings persistent.

Operation Mode
The device’s operation mode. Currently only “Modbus” is supported.

Modbus Slave Address
Set the Modbus slave address from 1…247. Default is 1.

Modbus Mode
The device’s modbus mode. Currently only “RTU/Binary” is supported.

Modbus Speed
This device supports multiple baud rates for Modbus communication.

All supported baud rates listed below. Only common baud rates can be saved persistent, un-common baud rates are only valid for the current runtime.

Common baud rates: 4800, 9600, 19200 (default), 38400, 56000, 57600, 115200

Un-common baud rates: 128000, 230400, 256000, 460800, 500000, 576000, 921000, 1000000, 1500000, 2000000, 3000000, 6000000, 12000000, 13500000

(For Hardware Revision 4 (MC4 0 x), only baud rates up to 500000 can be configured)

Modbus Termination Resistor Setting
A 120 Ohm termination resistor can be switched between the RS485 A and B lines inside this device. Available settings for the “Mb Term. Resistor” in the configuration register are “Off” and “On”. Default is “Off”.

Setting is available in hardware version 4 and later. For hardware version 3, only setting “Off” is available.

5.2 GPIO Channels

Two additional controllable input or output channels may be used to add external sensors or actors.

For electrical characteristics or wiring of the GPIO channels, please have a look at the datasheet of the smartPREDICT-AE RS485/USB.

In the current implementation, the GPIO channels can be turned off (high impedance), be a GPIO output from the Autonomous Condition Monitoring module or act as input to trigger the start and end of a measurement.

In future releases there are more possibilities like:

- PWM Input / Output
- ADC / DAC
- UART
- I2C
- CAN (needs external bus driver hardware)

5.2.1 GPIO channel implementation

In the GPIO channel implementation the modes as well as configuration of the GPIOs can be selected. The modes can be selected individually for both GPIOs.

Mode
The “Mode” setting of the configuration register allows to select between the different modes. Before using one of the GPIOs in an other module, the mode needs to be selected via the GPIO channel interface. For example, before using the “Continuous GPIO Triggered Start / Stop” of a measurement in the AE Sensor configuration registers, the GPIO Mode - for the GPIO to which the trigger signal will be applied - needs to be set to “Trigger Measurement Start / Stop”.

The following GPIO Modes are currently available:

- Off (high impedance)
- Autonomous Condition Monitor (GPIO Output) (default)
- Trigger Measurement Start / Stop (GPIO Input, ISR)
- Pulse counter per second (GPIO Input, ETR)

Mode “Pulse counter per second” is currently only available for GPIO 1.

Pull-Up / Pull-Down setting

Following “Pull-Up / Pull-Down” settings are available for the GPIOs:

- No-Pull
- Pull-Up
- Pull-Down (default)

The pull setting is not available for modes “Off” and “Autonomous Condition Monitor”. In this modes, the pull settings can’t be modified.

In mode “Trigger Measurement Start / Stop” only setting “Pull-Down is available”.

In mode “Pulse counter per second” all settings are available

High-Active / Low-Active

The “High-Active / Low-Active setting” for the GPIOs is currently only available for the GPIO Mode “Trigger Measurement Start / Stop”. In all other GPIO modes it’s a “don’t care” and can’t be modified.

In the mode “Trigger Measurement Start / Stop” the only possible setting is “High-Active”, which is selected by default. So only a positive impulses can currently trigger the start (rising edge) and stop (falling edge) of a measurement.

More settings will be added with future implementations.

Pulses per round

To measure a RPM (rounds per minute) pulse, the pulses per round setting is needed for the calculation and can be configured with this setting.

This setting is only available in mode “Pulse counter per second”.

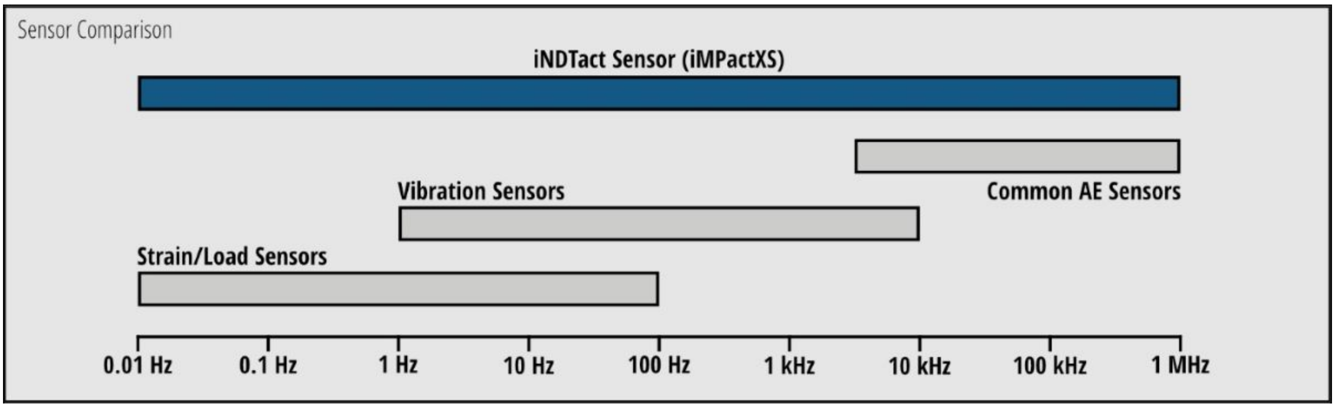
The resulting pulses per second and RPM can be read via the Modbus Measurement registers “Pulses per second” and “RPM rounds per minute” in the chapter “GPIO Channel”.

5.3 Acoustic Emission (AE) Sensor

The Acoustic Emission (AE) sensor is processed through an analog-to-digital converter system. A digital controllable filter pre-processes the signal before it is fed to the parameter calculation module.

The AE sensor, build into the smartPREDICT is an iNDTact **iMPactXS**. The iMPactXS is a high-performance acoustic emission and dynamic load sensor especially for structural health monitoring applications of lightweight structures (e.g. composites), turbines, machines, gears, bearings, electromechanical device, etc.

For sensor position & orientation have a look at the datasheet of the smartPREDICT-AE RS485/USB.



To get more information on the iMPactXS and its characteristics, have a look at the datasheet of the iNDTact iMPact XS.

The iMPactXS sensor is connected to an iNDTact **chAMP Head**. The chAMP Head is a stand-alone analog charge amplifier module with digital gain and cut-off frequency setup. The digital gain and cut-off frequency settings are described in more detail in the section “AE Sensor Analog Signal Filter” below. For more information on the chAMP Head, have a look at the datasheet of the iNDTact chAMP Head.

There is a different version of the smartPREDICT, where instead of the AE sensor, there is an analogue accelerometer (ADXL100x) connected to the analogue interface of the main sensor. Please have a look at chapter of accelerometer ADXL100x for more information.

5.3.1 AE Sensor Implementation

The AE sensor signal is fed to different function units, including - Parameter calculation - Autonomous condition monitoring - Digital signal filter - Data recorder - Power Spectral Density (PSD)

The implementation of these function units is described in the following sections and won’t be covered by the “AE Sensor Implementation”. This section will describe the configuration register settings of the mode, sample rate, analog filter frequency and analog filter gain of the AE sensor.

Changes of the AE sensor configuration register settings for mode, sample rate, analog filter frequency and analog filter gain are applied immediately and are valid for the current runtime. If you want to make these settings persistent, write coil “Save Configuration”.

Mode
The “Mode” setting of the configuration register for the AE sensor enables or disables the AE data acquisition and makes different modes available.

The following modes can be chosen

- Off
- Continuous Window Length
- Continuos GPIO Triggered Start / Stop

More modes like “Continuos SW Triggered Start / Stop” “Continuos SW Triggered Start” or “Continuos GPIO Triggered Start” as well as all modes as “Single Shot” will be added in the future.

In the “Continuous Window Length” Mode, data is processed continuously. Length of parameter calculation window can be selected via Modbus Configuration Register “Window Length”.

In the “Continuos GPIO Triggered Start / Stop” mode the parameter calculation start and end is completely triggered by an external signal, which is applied to the external GPIOs. The length of the last calculation window can be read via the Measurement Register “Elapsed Window Length”. Ensure, that you set at least one GPIO channel to the mode “Trigger Measurement Start / Stop” to use this AE Sensor Mode.

Sample Rate
Different AE sensor sample rates can be configured with the “Sample Rate” setting in the configuration registers of the AE sensor. It’s recommended to not change the sample rate of the AE sensor, except it’s needed for longer “Time to Record” in the data recorder module.

The following sample rates can be configured. The sample rate is based on the sample rate of an internal ADC and therefore it’s not an exact value. The theoretical and real sample rates can be viewed in the following table. There is also added the real sample rate as fraction, because the values in the table are rounded.

Theoretical sample rate	Real sample rate	Real sample rate (fraction)
8 kHz	8015.950521 Hz	98.5e6 / 12288 Hz
16 kHz	16031.90104 Hz	98.5e6 / 6144 Hz
32 kHz	32063.80208 Hz	98.5e6 / 3072 Hz
48 kHz	48095.70313 Hz	98.5e6 / 2048 Hz
64 kHz	64127.60417 Hz	98.5e6 / 1536 Hz
96 kHz	96191.40625 Hz	98.5e6 / 1024 Hz

Analog Filter Frequency
The “Analog Filter Frequency” can be configured in the configuration register of the AE sensor and can be one of the following values:

- Frequency Low
- Frequency Medium
- Frequency High

To get more detailed information on the this setting, have a look at the section “AE Sensor Analog Signal Filter” below.

Analog Filter Gain
The “Analog Filter Gain” can be configured in the configuration register of the AE sensor and can be one of the following values:

- Gain -20 dB
- Gain 0 dB
- Gain +20 dB
- Gain Max

To get more detailed information on the this setting, have a look at the section “AE Sensor Analog Signal Filter” below.

5.3.2 AE Sensor Default Values

The following table shows the default values of the AE sensor in the configuration register section.

AE Sensor	Default Value
Mode	1 Continuous
Sample Rate	8 96 kHz
Analog filter freq.	2 Medium
Analog filter gain	1 0 dB
Filter Type	0 Off
Filter Pass Type	0 Lowpass
Filter Order	1
Filter Lower Fc	100.0 (Hz)
Filter Upper Fc	200.0 (Hz)
Filter PB Ripple	1.0 (dB)
Filter SB Attenuation	60.0 (dB)
Window Length (s)	1.0 (s)
K(t) Ref. Abs. Peak	0.0
K(t) Ref. RMS	0.0
sPR Ref. Sample Rate	0 (Hz)
sPR Ref. Window Len.	NAN
sPR Tuning Low	0.0 (Hz)
sPR Tuning High	48000.0 (Hz)
sPRc Tuning Low	0.0 (Hz)
sPRc Tuning High	48000.0 (Hz)

5.3.3 AE Sensor Analog Signal Filter

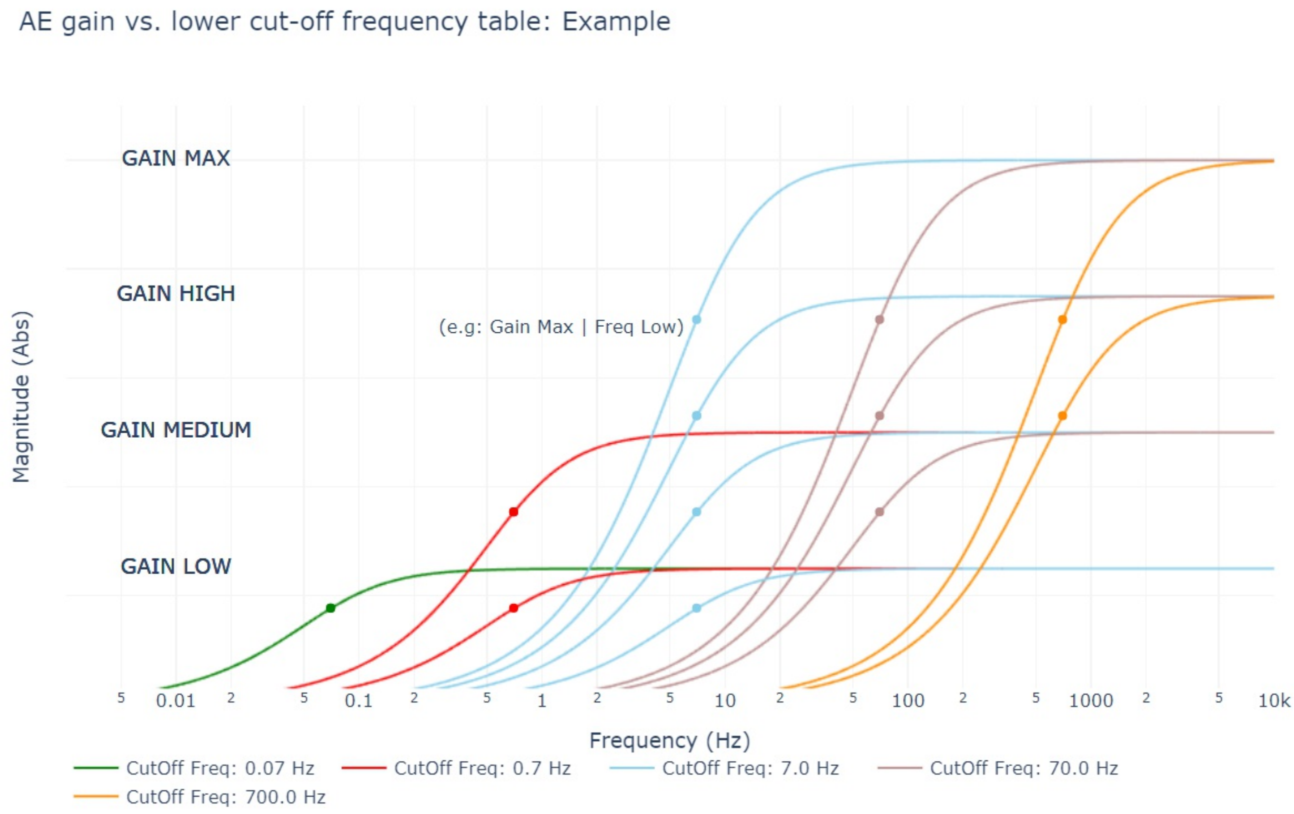
The AE sensor has an analog filter unit. The gain and frequency of this filter can be configured.

Default settings are: Frequency medium (Mid), Gain 0 dB

AE gain vs. lower cut-off frequency table

	Gain -20 dB	Gain 0 dB	Gain +20 dB	Gain Max
Frequency Low	0.07 Hz	0.7 Hz	7 Hz	> 7 Hz
Frequency Mid	0.7 Hz	7 Hz	70 Hz	> 70 Hz
Frequency High	7 Hz	70 Hz	700 Hz	> 700 Hz

Note: The gain "Max" setting may be used but neither the gain nor the cut-off frequency are guaranteed. Measurements from different devices are expected to have a high spread using this setting.



5.3.4 AE Sensor chAMP Signal Generator

The chAMP Head of the AE sensor supports a signal generator. The user can set different signal types and signal frequencies which can be configured with "Signal Type" and "Signal Frequency" in the "AE Sensor chAMP Signal Generator" section of the configuration registers.

The following signal types are supported:

- Off
- Sine
- Triangle
- Square
- Ramp

The signal frequency must be between 0.0 Hz and 423952.0 Hz.

Note: Maximum sample rate of AE sensor is 96 kHz, so it's not recommended to set at "Signal Frequency" above 48000 Hz

Default Values

chAMP Signal Generator	Default Value
Signal Type	0 Off
Signal Frequency	1000.0 (Hz)

5.3.5 Accelerometer ADXL100x

At the smartPREDICT-ID RS485/USB, instead of the AE sensor, there is an analogue accelerometer ADXL100x with +/-100g in Z-axis connected to the analogue sensor input of device.

If you are not sure which sensor is connected inside your smartPREDICT device, you can have a look at Modbus Configuration Register address 0x4000A (Main sensor HW). There could be the following values:

Main sensor Hardware	Value
chAMP (AE Sensor)	0
ADXL	1

All settings and default values which apply to the AE sensor, also apply to the ADXL100x. The exceptions to this are that the chAMP signal generator and the chAMP frequency- and gain-settings, which can’t be used at the ADXL100x (they are “don’t cares”).

5.4 Acceleration Sensor

The acceleration sensor can be configured in terms of sensitivity. The acceleration sensor provides a combined XYZ channel and each X, Y and Z channels (axes) as a single data source. A digital controllable filter pre-processes the signal before it is fed to the parameter calculation module.

The embedded acceleration sensor hardware depends on the hardware version of the device. Starting with hardware version 6 (MC6 0 0), there is a STMicroelectronics LSM6DSO32X integrated in the device. Starting with hardware version 7 (MC7 0 0), there is ANALOG DEVICES ADXL373 in addition to STMicroelectronics LSM6DSO32X integrated in the device. The acceleration data is read from the two MEMS depending upon the range. For legacy devices, there is a Bosch BMI160 implemented.

To check, which hardware is build in inside your smartPREDICT device, you can have a look at Modbus Configuration Register address 0x4000B (Acc/Gyro sensor HW). There could be the following values:

Acc/Gyro sensor Hardware	Value
Bosch BMI160	0
STMicroelectronics LSM6DSO32X	1
STMicroelectronics LSM6DSO32X and ANALOG DEVICES ADXL373	2

The inertial module (BMI160 / LSM6DSO32X) build into the smartPREDICT combining an accelerometer and a gyroscope. The inertial module is the data source for the different channels of the acceleration sensor.

For sensor position & orientation have a look at the datasheet of the smartPREDICT-AE RS485/USB.

The data of the combined XYZ channel of the accelerometer sensor is acquired by vector-addition:
XYZ = sqrt(X*X + Y*Y + Z*Z)

STMicroelectronics LSM6DSO32X

Following sensitivity (typ.) is set per range

Range	Sensitivity
+/-4 g	8192 LSB/g
+/-8 g	4096 LSB/g
+/-16 g	2048 LSB/g
+/-32 g	1024 LSB/g

The accelerometer sensor filter mode inside the LSM6DSO32X is configured to “high performance mode”. Although the hardware supports different filter modes, it’s not intended to change the filter mode in the future. - At a sample rate of 6667 Hz, this will result in a 3dB cut-off frequency at 3333 Hz.

To get more information on the accelerometer characteristics, have a look at the datasheet of the LSM6DSO32X.

Bosch BMI160

Following sensitivity (typ.) is set per range

Range	Sensitivity
+/-2 g	16384 LSB/g
+/-4 g	8192 LSB/g
+/-8 g	4096 LSB/g
+/-16 g	2048 LSB/g

The accelerometer sensor filter mode inside the BMI160 is configured to “normal mode”. Although the hardware supports different filter modes, it’s not intended to change the filter mode in the future. - At a sample rate of 1600 Hz, this will result in a 3dB cut-off frequency at 684 Hz.

To get more information on the accelerometer characteristics, have a look at the datasheet of the BMI160.

ANALOG DEVICES ADXL373

Following sensitivity (typ.) is set per range

Range	Sensitivity
+/-400 g	200 mg/LSB

The accelerometer sensor filter mode inside the ADXL373 is configured to “Full-Bandwidth measurement mode” with “low-noise operation mode”. Although the hardware supports different filter modes, it’s not intended to change the filter mode in the future. - At a sample rate of 5120 Hz, this will result in a 3dB cut-off frequency at 2560 Hz.

To get more information on the accelerometer characteristics, have a look at the datasheet of the ADXL373.

5.4.1 Acceleration Sensor Implementation

The acceleration sensor signal of the different channels (XYZ, X, Y, Z) is fed to different function units, including - Parameter calculation - Autonomous condition monitoring - Digital signal filter - Data recorder - Power Spectral Density (PSD)

The implementation of these function units is described in the following sections and won’t be covered by the “Acceleration Sensor Implementation”. This section will describe the configuration register settings of the mode, sample rate and range of the acceleration sensor channels.

Changes of the acceleration sensor configuration register settings for mode, sample rate and range of the different channels are applied immediately and are valid for the current runtime. If you want to make these settings persistent, write coil “Save Configuration”.

Mode
The “Mode” setting of the configuration register for the accelerometer sensor enables or disables the data acquisition of the selected accelerometer channel and makes different modes available.

The following modes can be chosen

- Off

- Continuous Window Length
- Continuos GPIO Triggered Start / Stop

More modes like “Continuos SW Triggered Start / Stop” “Continuos SW Triggered Start” or “Continuos GPIO Triggered Start” as well as all modes as “Single Shot” will be added in the future.

In the “Continuous Window Length” Mode, data is processed continuously. Length of parameter calculation window can be selected via Modbus Configuration Register “Window Length”.

In the “Continuos GPIO Triggered Start / Stop” mode the parameter calculation start and end is completely triggered by an external signal, which is applied to the external GPIOs. The length of the last calculation window can be read via the Measurement Register “Elapsed Window Length”. Ensure, that you set at least one GPIO channel to the mode “Trigger Measurement Start / Stop” to use this Accelerometer Mode.

Note: Changing the mode of one accelerometer sensor channel (XYZ, X, Y, Z) will change the mode of all channel.

Sample Rate
Currently only the sample rate of 1600 Hz / 6667 Hz is supported in the configuration register for the selected accelerometer channel. Although the hardware supports different (slower) sample rates, this is not intended to change in the future.

Range
The “Range” of the configuration register for the selected accelerometer sensor channel can be changed to the following values:

- +/-2 g (BMI160 only)
- +/-4 g
- +/-8 g
- +/-16 g
- +/-32 g (LSM6DSO32X only)
- +/-400 g (ADXL373 only)

Note: Changing the range of one accelerometer sensor channel (XYZ, X, Y, Z) will change the mode of all channel.

5.4.2 Acceleration Sensor Default Values

The default values are the same for each channel (XYZ, X, Y, Z) of the acceleration sensor. Both the MEMS LSM6DSO32X and ADXL373 operate at the sample rate of 6667 Hz and have the similar default values when used together.

STMicroelectronics LSM6DSO32X

Acceleration Sensor	Default Value
Mode	1 Continuous
Sample Rate	17 6667 Hz
Range	0 +/-4 g (LSM6DSO32X only)
Reserved	0

Acceleration Sensor	Default Value
Filter Type	0 Off
Filter Pass Type	0 Lowpass
Filter Order	1
Filter Lower Fc	100.0 (Hz)
Filter Upper Fc	200.0 (Hz)
Filter PB Ripple	1.0 (dB)
Filter SB Attenuation	60.0 (dB)
Window Length (s)	1.0 (s)
K(t) Ref. Abs. Peak	0.0
K(t) Ref. RMS	0.0
sPR Ref. Sample Rate	0 (Hz)
sPR Ref. Window Len.	NAN
sPR Tuning Low	0.0 (Hz)
sPR Tuning High	3333.3333 (Hz)
sPRc Tuning Low	0.0 (Hz)
sPRc Tuning High	3333.3333 (Hz)

Bosch BMI160

Acceleration Sensor	Default Value
Mode	1 Continuous
Sample Rate	7 1600 Hz
Range	0 +/-2 g
Reserved	0
Filter Type	0 Off
Filter Pass Type	0 Lowpass
Filter Order	1
Filter Lower Fc	100.0 (Hz)
Filter Upper Fc	200.0 (Hz)
Filter PB Ripple	1.0 (dB)
Filter SB Attenuation	60.0 (dB)
Window Length (s)	1.0 (s)

Acceleration Sensor	Default Value
K(t) Ref. Abs. Peak	0.0
K(t) Ref. RMS	0.0
sPR Ref. Sample Rate	0 (Hz)
sPR Ref. Window Len.	NAN
sPR Tuning Low	0.0 (Hz)
sPR Tuning High	800.0 (Hz)
sPRc Tuning Low	0.0 (Hz)
sPRc Tuning High	800.0 (Hz)

5.5 Gyroscope Sensor

The gyroscope sensor can be configured in terms of sensitivity. The gyroscope sensor provides a combined XYZ channel and each X, Y and Z channels (axes) as a single data source. A digital controllable filter pre-processes the signal before it is fed to the parameter calculation module.

The embedded gyroscope hardware depends on the hardware version of the device. Starting with hardware version 6 (MC6 0 0), there is a STMicroelectronics LSM6DSO32X integrated in the device. Starting with hardware version 7 (MC6 7 0), there is a ANALOG DEVICES ADXL373 in addition to STMicroelectronics LSM6DSO32X integrated in the device. The acceleration data is read from the two MEMS depending upon the range. But for gyroscope data only STMicroelectronics LSM6DSO32X is used. For legacy devices, there is a Bosch BMI160 implemented.

To check, which hardware is build in inside your smartPREDICT device, you can have a look at Modbus Configuration Register address 0x4000B (Acc/Gyro sensor HW). There could be the following values:

Acc/Gyro sensor Hardware	Value
Bosch BMI160	0
STMicroelectronics LSM6DSO32X	1
STMicroelectronics LSM6DSO32X and ANALOG DEVICES ADXL373	2

The inertial module (BMI160 / LSM6DSO32X) build into the smartPREDICT combining an accelerometer and a gyroscope. The inertial module is the data source for the different channels of the gyroscope.

For sensor position & orientation have a look at the datasheet of the smartPREDICT-AE RS485/USB.

The data of the combined XYZ channel of the gyroscope sensor is acquired by vector-addition:
XYZ = sqrt(X*X + Y*Y + Z*Z)

STMicroelectronics LSM6DSO32X

Following sensitivity (typ.) is set per range

Range	Sensitivity
+/-125 °/s	228.6 LSB/°/s
+/-250 °/s	114.3 LSB/°/s
+/-500 °/s	57.1 LSB/°/s
+/-1000 °/s	28.6 LSB/°/s
+/-2000 °/s	14.3 LSB/°/s

The gyroscope filter mode inside the LSM6DSO32X is configured to “high performance mode”. Although the hardware supports different filter modes, it’s not intended to change the filter mode in the future. - At a sample rate of 6667 Hz, this will result in a 3dB cut-off frequency at 1441.8 Hz.

To get more information on the gyroscope characteristics, have a look at the datasheet of the LSM6DSO32X.

Bosch BMI160

Following sensitivity (typ.) is set per range

Range	Sensitivity
+/-125 °/s	262.4 LSB/°/s
+/-250 °/s	131.2 LSB/°/s
+/-500 °/s	65.6 LSB/°/s
+/-1000 °/s	32.8 LSB/°/s
+/-2000 °/s	16.4 LSB/°/s

The gyroscope sensor filter mode inside the BMI160 is configured to “normal mode”. Although the hardware supports different filter modes, it’s not intended to change the filter mode in the future. At a sample rate of 1600 Hz, this will result in a 3dB cut-off frequency at 523.9 Hz.

To get more information on the gyroscope characteristics, have a look at the datasheet of the BMI160.

5.5.1 Gyroscope Sensor Implementation

The gyroscope sensor signal of the different channels (XYZ, X, Y, Z) is fed to different function units, including

- Parameter calculation
    - Autonomous condition monitoring
    - Digital signal filter
- Data recorder
- Power Spectral Density (PSD)

The implementation of these function units is described in the following sections and won’t be covered by the “Gyroscope Sensor Implementation”. This section will describe the configuration register settings of the mode, sample rate and range of the gyroscope sensor channels.

Changes of the gyroscope sensor configuration register settings for mode, sample rate and range of the different channels are applied immediately and are valid for the current runtime. If you want to make these settings persistent, write coil “Save Configuration”.

Mode
The “Mode” setting of the configuration register for the gyroscope sensor enables or disables the data acquisition of the selected gyroscope sensor channel and makes different modes available.

The following modes can be chosen

- Off
- Continuous Window Length
- Continuos GPIO Triggered Start / Stop

More modes like “Continuos SW Triggered Start / Stop” “Continuos SW Triggered Start” or “Continuos GPIO Triggered Start” as well as all modes as “Single Shot” will be added in the future.

In the “Continuous Window Length” Mode, data is processed continuously. Length of parameter calculation window can be selected via Modbus Configuration Register “Window Length”.

In the “Continuos GPIO Triggered Start / Stop” mode the parameter calculation start and end is completely triggered by an external signal, which is applied to the external GPIOs. The length of the last calculation window can be read via the Measurement Register “Elapsed Window Length”. Ensure, that you set at least one GPIO channel to the mode “Trigger Measurement Start / Stop” to use this Gyroscope Mode.

Note: Changing the mode of one gyroscope sensor channel (XYZ, X, Y, Z) will change the mode of all channel.

Sample Rate
Currently only the sample rate of 1600 Hz / 6667 Hz is supported in the configuration register for the selected gyroscope channel. Although the hardware supports different (slower) sample rates, this is not intended to change in the future.

Range
The “Range” of the configuration register for the selected gyroscope sensor channel can be changed to the following values:

- +/-125 °/s
- +/-250 °/s
- +/-500 °/s
- +/-1000 °/s
- +/-2000 °/s

Note: Changing the range of one gyroscope sensor channel (XYZ, X, Y, Z) will change the range of all channel.

5.5.2 Gyroscope Sensor Default Values

The default values are the same for each channel (XYZ, X, Y, Z) of the gyroscope sensor.

STMicroelectronics LSM6DSO32X

Acceleration Sensor	Default Value
Mode	1 Continuous
Sample Rate	17 6667 Hz
Range	0 +/-125 °/s
Reserved	0
Filter Type	0 Off
Filter Pass Type	0 Lowpass
Filter Order	1
Filter Lower Fc	100.0 (Hz)
Filter Upper Fc	200.0 (Hz)
Filter PB Ripple	1.0 (dB)

Acceleration Sensor	Default Value
Filter SB Attenuation	60.0 (dB)
Window Length (s)	1.0 (s)
K(t) Ref. Abs. Peak	0.0
K(t) Ref. RMS	0.0
sPR Ref. Sample Rate	0 (Hz)
sPR Ref. Window Len.	NAN
sPR Tuning Low	0.0 (Hz)
sPR Tuning High	3333.3333 (Hz)
sPRc Tuning Low	0.0 (Hz)
sPRc Tuning High	3333.3333 (Hz)

Bosch BMI160

Gyroscope Sensor	Default Value
Mode	1 Continuous
sample rate	6 1600 Hz
Range	0 +/-125 °/s
Reserved	0
Filter Type	0 Off
Filter Pass Type	0 Lowpass
Filter Order	1
Filter Lower Fc	100.0 (Hz)
Filter Upper Fc	200.0 (Hz)
Filter PB Ripple	1.0 (dB)
Filter SB Attenuation	60.0 (dB)
Window Length (s)	1.0 (s)
K(t) Ref. Abs. Peak	0.0
K(t) Ref. RMS	0.0
sPR Ref. Sample Rate	0 (Hz)
sPR Ref. Window Len.	NAN
sPR Tuning Low	0.0 (Hz)
sPR Tuning High	800.0 (Hz)

Gyroscope Sensor	Default Value
sPRc Tuning Low	0.0 (Hz)
sPRc Tuning High	800.0 (Hz)

5.6 Autonomous Condition Monitoring

Five independent threshold / range monitors can be configured. Their alert outputs can be combined with three logic modules to create more complex condition monitoring alerts. The alerts can be accessed via the status coil registers and can be fed to the USER LED or to a GPIO output.

The conditions are checked each time a window of the vibration sensor channel has been expired and new data is ready.

If you configure a condition monitor module or a condition monitor logic module, the configuration is only valid during the current runtime. Write coil “Save Configuration” to make current configuration persistent.

For configuration and alert output, only the configuration register and status coil register sections are used. Measurement registers aren’t used for autonomous condition monitoring.

5.6.1 Condition Monitoring Modules

The five independent condition monitoring modules can be configured in the following way.

Mode
The following modes can be set via the configuration register “Mode” of the selected condition monitoring channel:

Mode	Description
Off	Condition monitor is not active
Above value	Alert will be triggered if actual value is above threshold value
Below value	Alert will be triggered if actual value is below threshold value
In range	Alert will be triggered if actual value is in the threshold range
Out of range	Alert will be triggered if actual value is out of the threshold range

Data Source Register
You can set any existing address from the measurement registers section as data source for the condition monitoring modules via the configuration register “Data Source Register” of the selected condition monitoring channel.
If the value to monitor is data type float32 or uint32, selected MSB (= first) address as data source register.

Lower Threshold, Upper Threshold
For mode “Above Value” set threshold value in the configuration register “Upper Threshold” of the selected condition monitoring channel. The “Lower Threshold” value will be ignored.

For mode “Below Value” set threshold value in the configuration register “Lower Threshold” of the selected condition monitoring channel. The “Upper Threshold” value will be ignored.

For modes “In range” and “Out of range” set lower und upper threshold value of the range in the configuration registers “Lower Threshold” and “Upper Threshold” of the selected condition monitoring channel.

When an (u)int16 measurement register is used as data source register, an (u)int16 threshold must be configured. The uint16 threshold is the MSB (=first) part of the Lower/Upper Threshold value.

When data type is uint32/float32 both register addresses are used.

Alert Output
If an alert is active, the status coil register of the selected condition monitor “CM x Alert” is set to ‘1’.
Additional to the status coil register the alert can be fed to the following hardware.

Alert Output	Description
Off	No additional alert output configured
USER LED	USER LED will display alert: Green - no alert, red - alert
GPIO Pin 1	GPIO Pin 1 will be set according to “Output Type” and “GPIO Mode”
GPIO Pin 2	GPIO Pin 2 will be set according to “Output Type” and “GPIO Mode”

Ensure, that you set the GPIO 1 or 2 to the mode “Autonomous Condition Monitor” in the GPIO Channels module to use the GPIOs

Output Type
Output types can be configured as “Switched” or “Latched” via configuration register “Output Type” of selected condition monitoring channel. - If output type is set to “Switched”, the alert output will automatically reset if alert is not active anymore - If output type is set to “Latched”, the alert output will stay active, even if alert is not active anymore - You can reset a “Latched” alert by writing ‘0’ to the status coil register “Condition Monitoring Alert” of the selected channel

GPIO Mode
If the alert output is configured as “GPIO Pin 1” or “GPIO Pin 2”, you can set the GPIO mode to “High Active” or “Low Active”.

First Alert Index
When an alert is activated for the first time, the device will save the current “AE Measurement Index” (independent of the actual data source register).
With knowledge of the current AE window length, the user can approximately calculate the time since the alert has been activated for the first time.
The first alert index can be read via the configuration register “First Alert Index” of the selected condition monitoring channel.

A status coil register flag “Condition Monitor First Alert” is provided to indicate if a “First Alert Index” is saved for the selected condition monitor channel. This flag is low-active. It will be ‘0’ if the “First Alert Index” is saved.
You can write ‘1’ to this coil to reset the flag. This will also reset the “First Alert Index” in the configuration register of the selected condition monitor channel.

First Alert Value
When an alert is activated for the first time, the device will save the current measurement register value, which triggered the alert. The value will be in the same format as the monitored measurement register: int16, uint16, uint32, float32
The first alert value can be read via the configuration register “First Alert Value” of the selected condition monitoring channel.

A status coil register flag “Condition Monitor First Alert” is provided to indicate if a “First Alert Value” is saved for selected condition monitor channel. This flag is low-active. It will be ‘0’ if the “First Alert Value” is

saved.
You can write ‘1’ to this coil to reset the flag. This will also reset the “First Alert Value” in the configuration register of the selected condition monitor channel.

5.6.2 Condition Monitoring Logic Modules

The condition monitoring logic modules combine two status coil registers with logic bit-operations to generate more complex alert outputs. The condition monitoring logic module is intended to be used with the status coil registers of the five condition monitoring alerts, but can also be used with any existing status coil register.
The three condition monitoring logic modules for complex condition monitoring alerts can be configured in the following way.

Mode

The following modes can be set via the configuration register “Mode” of the selected condition monitoring logic channel:

Mode	Description
Off	Condition monitor logic not active
AND	AND operation is performed
OR	OR operation is performed
XOR	Exclusive-OR operation is performed
NOR	NOT-OR operation is performed
XNOR	Exclusive-NOR operation is performed
NAND	NOT-AND operations is performed

Truth table comparison of the logic module modes

Input A	Input B	AND	OR	XOR	NOR	XNOR	NAND
0	**0**	0	0	0	1	1	1
0	**1**	0	1	1	0	0	1
1	**0**	0	1	1	0	0	1
1	**1**	1	1	0	0	1	0

Source Registers

Select any address from the status coil registers section as Input A or B to the logic module.
The two data source registers can be configured in the configuration register section of the selected condition monitoring logic channel.

Suggestion: When using combination between CM logic x’s configure the output at CM logic C.

Alert Output If an alert is active, the status coil register of the selected condition monitor “CM x Alert” is set to ‘1’. Additional to the status coil register the alert can be fed to the following hardware.

Alert Output	Description
Off	No additional alert output configured
USER LED	USER LED will display alert: Green - no alert, red - alert
GPIO Pin 1	GPIO Pin 1 will be set according to "Output Type" and "GPIO Mode"
GPIO Pin 2	GPIO Pin 2 will be set according to "Output Type" and "GPIO Mode"

Ensure, that you set the GPIO 1 or 2 to the mode "Autonomous Condition Monitor" in the GPIO Channels module to use the GPIOs

Output Type
Output types can be configured as "Switched" or "Latched" via configuration register "Output Type" of selected condition monitoring logic channel.

- If output type is set to "Switched", alert output will automatically reset if alert is not active anymore
- If output type is set to "Latched", alert output will stay active, even if alert is not active anymore
    - You can reset a "Latched" alert by writing '0' to status coil register "Condition Monitoring Logic Alert" of selected channel

GPIO Mode
If alert output is configured as "GPIO Pin 1" or "GPIO Pin 2", you can set the GPIO mode to "High Active" or "Low Active"

First Alert Index
When an alert is activated for the first time, the device will save the current "AE Measurement Index" (independent of the actual data source register).
With knowledge of the current AE window length, the user can approximately calculate the time since the alert has been activated for the first time.
The first alert index can be read via the configuration register "First Alert Index" of the selected condition monitoring logic channel.

A status coil register flag "Condition Monitor Logic First Alert" is provided to indicate if a "First Alert Index" is saved for selected condition monitor logic channel. This flag is low-active. It will be '0' if the "First Alert Index" is saved.
You can write '1' to this coil to reset the flag. This will also reset the "First Alert Index" in the configuration register of the selected condition monitor logic channel.

5.6.3 Autonomous Condition Monitoring Default Settings

Default settings of the configuration register values for the condition monitor module

CM Module	Default Value
Mode	0 Off
Data Source Register	0x0000
Lower Threshold	0.0
Upper Threshold	0.0

CM Module	Default Value
Alert Output	0 Off
Output Type	0 Switched
GPIO Mode	0 High Active
First Alert Index	0
First Alert Value	0.0

Default settings of the configuration register values for the condition monitor logic module

CM Logic Module	Default Value
Mode	0 Off
Source Register 1	0x0000
Source Register 2	0x0000
Alert Output	0 Off
Output Type	0 Switched
GPIO Mode	0 High Active
First Alert Index	0

5.7 Data Recorder

The data recorder can be used to collect sample data for more complex analytics. The recorded data is saved in the smartPREDICT's SDRAM. Currently two types of data can be recorded:

- Raw data of a vibration sensor channel
  - No signal filtering or other data processing is applied to this data. It's the raw data of the vibration sensor channel, converted from the ADC and multiplied with the selected unit conversion factor
- Power Spectral Density (PSD) array of vibration sensor channel
  - For more information on the PSD calculation look at section "Power Spectral Density (PSD) calculation"

Note: The Data Recorder is not conform with the Modbus specification. Modbus is not designed to read one register multiple times to get different values, which is how the Data Recorder is implemented.

5.7.1 Data Recorder Implementation

5.7.1.1 Record Types

The following record types can be set via the configuration register "Record Type"

Record Type	Vibration Sensor Channel
Raw Data	AE sensor
Raw Data	(combined) AE sensor, accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
Raw Data	(combined) accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
Raw Data	(combined) accelerometer all channel (XYZ, X, Y, Z)
Raw Data	Accelerometer XYZ channel
Raw Data	Accelerometer X channel
Raw Data	Accelerometer Y channel
Raw Data	Accelerometer Z channel
Raw Data	(combined) gyroscope all channel (XYZ, X, Y, Z)
Raw Data	Gyroscope XYZ channel
Raw Data	Gyroscope X channel
Raw Data	Gyroscope Y channel
Raw Data	Gyroscope Z channel
PSD array	AE sensor
PSD array	Accelerometer XYZ channel
PSD array	Accelerometer X channel
PSD array	Accelerometer Y channel

Record Type	Vibration Sensor Channel
PSD array	Accelerometer Z channel
PSD array	Gyroscope XYZ channel
PSD array	Gyroscope X channel
PSD array	Gyroscope Y channel
PSD array	Gyroscope Z channel

5.7.1.2 Data[n]

Access to the data is granted by the measurement registers “Data0” to “Data61” in the data recorder section. This registers represent 62 Samples of (float32) data. The content of the “Data[n]” registers gets automatically updated if a “Data0” register is read.
It doesn’t matter if you read the data sample by sample from Data0 register or if you read more samples with one command, for example read 64 registers (which represents 32 samples: Data0 to Data31) with Modbus command 0x04 read input registers.

If multiple registers of data are read via Modbus command 0x04 read input registers, Data0 register must be included, so that index of Data0 register matches to the read data samples.

5.7.1.3 Index of Data0 and Record Index

The index of Data0 if provided via measurement register “Index of Data0”. Index n of Data[n] is the offset to the index of Data0.
The index of Data0 is incremented every time, the MSB register of Data[n] is read, so it always represents the index, when Data0 is read the next time. If data is lost, the index of Data0 can be set by the configuration register address “Record Index”. If multiple measurement registers of data are read via Modbus Command “0x04 Read Input Registers”, “Data0” register must be included, so that “Index of Data0” register matches to the read data samples and content of Data[n] gets updated.

5.7.1.4 Time reaming and Samples remaining

The measurement register values “Time remaining” and “Samples remaining” can be used to check, how much time / samples is / are remaining till end of recording. If recording hasn’t been started, these values are the total time / samples to record. If recording is finished, these values are ‘0’.

5.7.1.5 Time recorded and Samples recorded

The measurement register values “Time recorded” and “Samples recorded” can be used to check, how much time / samples is / are recorded since start of recording. If recording hasn’t been started, these values are ‘0’. If recording is finished, these values are equal to total time / samples recorded.

5.7.1.6 Start Recording

To Start the recoding of the configured data, write write ‘1’ to configuration register “Start Recording”. Values “Record Type”, “Time to Record” and “Samples to Record” must be set before recording is started.

This register stays at ‘1’ as long as recording is ongoing and resets to ‘0’ if recording is finished. While recording is running, no new recording can be started and “Record Type”, “Time to Record” and “Samples to Record” can’t be changed. If you need to change the configuration, write ‘1’ to configuration register “Stop Recording” to stop current recording

5.7.1.7 Stop Recording

Write ‘1’ to configuration register “Stop recording” to stop current recoding. If you stop a recording, already recorded data can be read.

5.7.1.8 Time to Record and Samples to Record

The configuration register values “Time to Record” and “Samples to Record” must be set before start of the recording. The meaning of these two values depend on the “Record Type” in the following way:

Record Type of Raw Data:

- “Time to Record” and “Samples to Record” depend on each other. Only one of the parameters needs to be set, the other one is set automatically
- Parameter for single raw data sources
    - “Record Types”:
        - AE sensor
        - Accelerometer XYZ channel
        - Accelerometer X channel
        - Accelerometer Y channel
        - Accelerometer Z channel
        - Gyroscope XYZ channel
        - Gyroscope X channel
        - Gyroscope Y channel
        - Gyroscope Z channel
    - Max “Samples to Record”: 8388608
    - Max “Time to Record”: Max “Samples To Record” / sample rate (eg. AE sensor: 8388608 / 96000 = 87.38 (s))
- Parameter for combined raw data sources
    - “Record Types”:
        - (combined) AE sensor, accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
        - (combined) Accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
        - (combined) Accelerometer all channel (XYZ, X, Y, Z)
        - (combined) Gyroscope all channel (XYZ, X, Y, Z)
    - It is **highly recommended** to set the parameter for the combined raw data record types via **“Time to Record”**
    - “Time to Record” is “Time to Record” for each data source of combined record type
    - “Samples to Record” is combined number of samples to record and depends on sample rates of data sources and “Time to Record”
    - Example:
        - “Record Type” - Raw data (combined) AE sensor, accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
        - Sample rates: AE sensor: 96000 Hz, accelerometer (Bosch BMI160): 1600 Hz,

gyroscope (Bosch BMI160): 1600 Hz
  - "Time to Record": 0.1000
  - "Samples to Record" = fs(AE) * "Time to Record" + 4 * fs(Acc) * "Time to Record" + 4 * fs(Gyro) * "Time to Record"
  - "Samples to Record" = 96000 * 0.1 + 4 * 1600 * 0.1 + 4 * 1600 * 0.1
  - "Samples to Record" = 10880
- Recorded raw data will be saved in complete blocks of each data source
- Example output for the above example:
  - Data[00000] … Data[09599]: 9600 Samples of raw data AE sensor
  - Data[09600] … Data[09759]: 160 Samples of raw data accelerometer XYZ channel
  - Data[09760] … Data[09919]: 160 Samples of raw data accelerometer X channel
  - Data[09920] … Data[10079]: 160 Samples of raw data accelerometer Y channel
  - Data[10080] … Data[10239]: 160 Samples of raw data accelerometer Z channel
  - Data[10240] … Data[10399]: 160 Samples of raw data gyroscope XYZ channel
  - Data[10400] … Data[10459]: 160 Samples of raw data gyroscope X channel
  - Data[10560] … Data[10719]: 160 Samples of raw data gyroscope Y channel
  - Data[10720] … Data[10879]: 160 Samples of raw data gyroscope Z channel

Record Type of PSD Array: - "Time to Record" and "Samples to Record" can't be set for PSD array record types and are pre-set by data recorder module if record type PSD array gets selected - Both depend on sample rate and window length for selected sensor data source of PSD array - "Time to Record" is the frequency resolution df of the PSD calculation - "Samples to Record" is the block length (number of input samples) of the PSD

5.7.2 Data Recorder Default Values

Default values of the configuration registers of the data recorder

Data Recorder	Default Value
Record Type	0x0001 Raw Data AE Sensor
Time to Record	87.3813 (s)
Samples to Record	8388608
Record Index	0
Start Recording	0
Stop Recording	0

Default values of the measurement registers of the data recorder

Data Recorder	Default Value
Time remaining	87.3813 (s)
Samples remaining	8388608
Time recorded	0.0 (s)
Samples recorded	0
Index of Data0	0

Data Recorder	Default Value
Data[0]	0.0

5.8 Digital Signal Filtering

Each vibration sensor has a controllable digital IIR signal filter for adjustment of the spectral content.

5.8.1 IIR Filter Specification

The following design parameters can be set by the user and describe the IIR filter.

Topology (Filter type) determines the design type of the filter.

Topologies:

- Off
- Butterworth
- Chebyshev
- Inverse Chebyshev*
- Elliptic
- Bessel

*(Inverse Chebyshev) is currently not available and will be included in a future update.

Pass Type specifies the passband of the filter according to the following values.

Pass types:

- Lowpass
- Highpass
- Bandpass
- Notch

Order is the filter order. Order should be between 1…12

Lower Fc is the low cut-off frequency and must observe the Nyquist criterion. The Nyquist criterion is 0 < Fc < 0.5*fs, where Fc is the cut-off frequency, and fs is the sampling frequency.

Lower Fc is the center frequency for pass types lowpass and highpass and the lower corner cut-off frequency for bandpass and notch.

Upper Fc is the high cut-off frequency. This parameter is only used when pass type is bandpass or notch.

Pb Ripple is the passband ripple. The Pb ripple must be between 0.0001 and 1.0 and is expressed in decibels.

Pb ripple is only used when topology is Chebyshev or Elliptic.

Sb Attenuation specifies the stopband attenuation. Sb attenuation must be between 20 and 120 and is expressed in decibels.

Sb attenuation is is only used when topology is Inverse Chebyshev or Elliptic.

5.8.2 Digital IIR Filter Implementation

All above described IIR filter specifications can be set via the configuration registers of the selected vibration sensor channel.

When you change one of the runtime parameter, the changes are not applied immediately. Write coil “Update Dig. Filter” of selected vibration sensor channel to apply changes.

Changes to digital signal filter are only valid during current runtime. To make configuration persistent, write coil “Save Configuration”.

5.8.3 IIR Filter Default Settings and Range

The following table shows the default settings and the range of the above described filter parameter.

Filter Parameter	Default Value	Allowed Range
Topology	0 Off	0, 1, 2, 4, 5
Pass Type	0 Lowpass	0…3
Order	1	1…12
Lower Fc	100.0 (Hz)	0…fs/2
Upper Fc	200.0 (Hz)	Lower Fc…fs/2
Pb Ripple	1.0 (dB)	0.0001…1.0
Sb Attenuation	60.0 (dB)	20.0…120.0

If user configures values which are out of range, it can lead to one of the following behaviors:

- Modbus error: A Modbus “Invalid Data Value” error is returned for the following parameters
    - Topology
    - Pass Type
    - Order
- Internal error: If user configures value which is out of range, this will be detected internally and the value will be reset to the default value for the following parameters
    - Lower Fc
    - Upper Fc
    - Pb Ripple
    - Sb Attenuation

5.9 Parameter Calculation

Each vibration sensor has a parameter calculation module. This units collect new data during a fixed duration (window length). The window length can be set by writing configuration register value “Window Length (s)” of the selected vibration sensor channel.

As soon as the collection is finished, new parameter outputs are calculated. This process runs continuous. When new data is ready, an output flag is provided as Modbus coil “Data Ready”. The coil “Data Ready” auto-resets to ‘0’ as soon as it’s ‘1’ and is read via Modbus command. So you can always poll this flag for the selected vibration sensor channel till it is set, read out the selected parameter from the channel, then poll the flag again to wait for new parameter.

You can (re-)start / synchronize a measurement by writing coil “Synchronize Trigger” for selected vibration sensor channel. You can also (re-)start / synchronize a measurement for all vibration sensor channel by writing the coil “Synchronize Trigger” in the general device information section of the status coil registers.

5.9.1 Provided Parameters

Parameter	Description
Range	Absolute range from min. to max. sample value. Also known as peak to peak value.
Absolute Peak	Maximum absolute sample value of all observed samples
Mean	Arithmetic mean of all observed sample values
RMS	Root-Mean-Square: Quadratic mean of all observed sample values
ZCR	Zero Crossing Rate: Rate of sign-changes along a signal per second (Usage: Determine dominant frequency)
Crest Factor	Ratio between maximum absolute peak and RMS
K(t)	Value defined in VDI 3832 (Usage: Monitoring of rotating machines), see section “K(t) calculation” below
SNR	Signal to noise ratio
Skewness	Measure of the asymmetry of the observed sample distribution
Kurtosis*	Measure of the tailedness of the observed sample distribution
sPR	Detection of vibration abnormalities, see section “sPR calculation” below
sPRs	Detection of vibration abnormalities, see section “sPR calculation” below
sPRc	Detection of vibration abnormalities, see section “sPR calculation” below
sPRsc	Detection of vibration abnormalities, see section “sPR calculation” below
Elapsed Win Len	Length of last calculation window

*(Kurtosis) Only correct if signal is free of any offset (“DC-free”). This applies to all iNDTact AE sensors and the gyroscope but not the acceleration sensor.

Note: Calculation of SNR is subjected to change with a future update and probably not correct. Calculations of Skewness and Kurtosis are also not subjected to change, but current implementation is not wrong. Formulas, how the parameters are calculated, will be added with a future update of this document.

5.9.2 16-bit (legacy) parameter

For each provided float32 parameter, there is a 16-bit (legacy) values provided for systems which can't handle float32 data types.

Registers between two 16-bit registers read as ‘0’. A factor to calculate the related float32 register value is provided for each 16-bit register.

5.9.3 K(t) calculation

Cite from the VDI3832, Chapter 8.2.1.5 K(t) value:

> The K(t) value is calculated from the root-mean-square and maximum absolute values of a broadband time signal of the vibratory acceleration. To do so, the ratio is formed between, on the hand, the current root-mean-square and maximum absolute values and, on the other hand, the values which were measured as reference values at the start or at a suitable time subsequently.
>
> K(t) = (a_rms(0) * a_p(0)) / (a_rms(t) * a_p(t))
> where
> a_rms(0) root-mean-square of the reference value
> a_p(0) maximum absolute value of the reference value
> a_rms(t) current root-mean-square value
> a_p(t) current maximum absolute value
>
> The reference values should be measured soon after the bearing goes into operation and if at all possible after a certain running-in period has elapsed. For this reference timepoint the K(t) value is equal to one. As wear on the bearing progresses – in other words, raceway damage – the K(t) value falls. The K(t) parameter can even increase a very little during the running-in period. In this way the K(t) value allows condition classification to be divided into the three classes:
>
> - undamaged
> - incipient damage
> - marked damage
>
> By linking the root-mean-square and maximum absolute values by multiplication, the K(t) value indicates changes in the time signal of acceleration. These changes are caused not only by shock excitations in the case of local damage (local pitting damage) but also by a general rise in structure-borne sound excitation in the case of distributed damage (raceway flaking, raceway corrosion, rolling element wear) and deficient lubrication.
>
> […]

5.9.3.1 K(t) Implementation

To use the K(t) value of a vibration sensor channel, you need to set the reference values a_rms(0) and a_p(0). This can be done by two ways:

- Write coil “Use K(t) Ref Values”
  - This will copy the current RMS and absolute peak runtime values into the K(t) RMS and absolute peak reference registers (in the configuration register section)

- Directly set the “K(t) Ref. RMS” and “K(t) Ref. Abs. Peak” values in the configuration register section.
  - This is especially useful, if you need to replace a device and and want to use the reference values of the previous device

Additional Information:

- To measure the reference values, bring the machine, to which the smartPREDICT is connected, in a normal state - this should be done soon after the machine goes into operation
- If you set new reference values, they are applied immediately and are only valid for current runtime
- Set coil “Save Configuration” to make setting persistent
- If no reference values are set, K(t) parameter in measurement registers is 0.0

5.9.4 sPR calculation

In combination with iNDTact’s unique high-performance sensors, our condition parameter sPR allows for an ultra-precise and robust trending, hence for super-early detection of vibration abnormalities in comparison to an initially recorded reference.

With its unique information density, the sPR parameter makes the detection of wear, degradation and instability of bearings as well as of gears or machinery operations a children’s game.

Recommended initial thresholds for most use cases:

sPR	Description
1	Indicates a state according to the initial condition
2*	Indicates slight beginning of degradation (warning)
2…10*	Indicates increasing degradation progress (alarm)
10*	Indicates rapid degradation and the quick approach towards total failure

*may be customized

An additional sPR-vector (implemented on request) also allows for detailed root cause analysis.

5.9.4.1 sPR Implementation

The sPR calculation is done for each vibration sensor channel individually and the sPR parameter can be accessed via measurement registers.

For the sPR calculation a reference needs to be saved in a normal state of the device.

- Bring the machine, to which the smartPREDICT is attached, in a normal state - this should be done soon after the machine goes into operation
- Write coil “Use sPR Ref Values” to use calculation of current window as reference for sPR calculation
- If reference is ready, coil “sPR ref values ready” will be set
- Sample rate and window length of the vibration sensor channel, at the time when the reference is set, will be saved at the configuration register values “sPR Ref. Sample Rate” and “sPR Ref. Window Len.”
- Reference is only valid during current runtime. Write coil “Save Configuration” to make sPR

reference persistent

Additional information on the sPR calculation

- If new sPR parameter is ready to be read, coil “Data Ready” of vibration sensor channel will be set to ‘1’
- If you want to start sPR calculation at a specific time, write coil “Synchronize Trigger” of vibration sensor channel
- If no sPR reference is provided, sPR parameter will be NAN and coil “sPR ref values ready” will be set to ‘0’
- If current sample rate or window length is different to sample rate or window length of sPR reference, there is a chance of sPR can’t be calculated
    - In this case coil “sPR Ref Values Ready” is reset to ‘0’ and sPR parameter is not calculated
- To delete sPR reference, write coil “Factory Reset”, this will delete the sPR reference on current runtime
    - To delete sPR reference persistent, write coil “Factory Reset” and “Save Configuration” afterwards
- You can can monitor a selected frequency window with the sPR calculation, by setting “sPR Tuning Low” and “sPR Tuning High” in the configuration register of the selected vibration sensor channel
    - “sPR Tuning Low”: Default value: 0.0 (Hz), Range: 0.0 … Sample Rate / 2
    - “sPR Tuning High”: Default value: Sample Rate / 2 (Hz), Range: “sPR Tuning Low” … Sample Rate / 2

5.10 Power Spectral Density (PSD) calculation

A Power Spectral Density (PSD) is calculated for each vibration sensor channel. The PSD is calculated several times and averaged during current window length of parameter calculation.

The PSD can be read via the data recorder.

The relevant parameters of the PSD calculation are depending on the sample rate and window length of each vibration sensor channel and are shown in the following tables.
f_s - Sample Rate (1/s); BL - Block Length; D - Duration (s); df - frequency-resolution (1/s)

When reading the PSD with the Data Recorder, the Modbus Measurement Registers “PSD df” provides the df frequency-resolution for the currently read PSD.

PSD parameter of AE Sensor

PSD parameter depend on the sample rate of the AE sensor

f_s	BL	D	df
8000	1024	0.128	7.8125
16000	2048	0.128	7.8125
32000	4096	0.128	7.8125
48000	4096	0.08533	11.71875
64000	4096	0.064	15.625
96000	4096	0.04267	23.4375

PSD parameter of Accelerometer

PSD parameter depend on the window length (= Duration D of PSD) of the accelerometer channel.

PSD parameter also depend on the sample rate of the used accelerometer hardware. For Bosch BMI160 sample rate is 1600 Hz, for LSM6DSO32X and ADXL373 sample rate is 6667 Hz.

f_s	BL	D	df
6667	32	0.0048	208.344
6667	64	0.0096	104.172
6667	128	0.0192	52.086
6667	256	0.0384	26.043
6667	512	0.0768	13.021
6667	1024	0.1536	6.511
6667	2048	0.3072	3.255

f_s	BL	D	df
1600	32	0.02	50

f_s	BL	D	df
1600	64	0.04	25
1600	128	0.08	12.5
1600	256	0.16	6.25
1600	512	0.32	3.125
1600	1024	0.64	1.5625
1600	2048	1.28	0.78125

PSD parameter of Gyroscope

PSD parameter depend on the window length (= Duration D of PSD) of the gyroscope channel.

PSD parameter also depend on the sample rate of the used gyroscope hardware. For Bosch BMI160 sample rate is 1600 Hz, for LSM6DSO32X sample rate is 6667 Hz.

f_s	BL	D	df
6667	32	0.0048	208.344
6667	64	0.0096	104.172
6667	128	0.0192	52.086
6667	256	0.0384	26.043
6667	512	0.0768	13.021
6667	1024	0.1536	6.511
6667	2048	0.3072	3.255

f_s	BL	D	df
1600	32	0.02	50
1600	64	0.04	25
1600	128	0.08	12.5
1600	256	0.16	6.25
1600	512	0.32	3.125
1600	1024	0.64	1.5625
1600	2048	1.28	0.78125

5.10.1 Order Analysis

With the PSD and information from a RPM (rounds per minute) signal, an order analysis can be done. The order analysis is the PSD with a scaled x-axis. In this case the x-axis is not df (frequency resolution), it's multiple of RPM. Therefore the smartPREDICT device offers a “dn” measurement which can be read via Modbus Measurement Register “PSD dn” in chapter “Data Recorder”.

“dn” is defined the following way:

dn = df / RPM (1/s)

To use the the RPM for the order analysis, the GPIO channel needs to be configured to read a pulse signal, which calculates the RPM. Configure Modbus Configuration register for GPIO channels the GPIO mode to “Pulse counter per second” and set the “Pulses per round” setting.

After this configuration, when reading the PSD with the Data Recorder, the “PSD dn” of the Data Recorder is available.

6 Supported Registers

Data transfer from and to registers is organized in terms of 16-bit integers. Please note: All registers in this manual are printed as absolute addresses. In general Modbus libraries only ask for the relative address according to the function code used.

6.1 Memory Layout

The virtual memory provided as Modbus registers is divided into three sections:

Reg Addresses	Description	Type	R/W	Notes
0x00000 - 0x0FFFF	Runtime Flags	bool	R/W	FC 0x01, 0x05
0x30000 - 0x3FFFF	Measurement	u16	R	FC 0x04
0x40000 - 0x4FFFF	Configuration	u16	R/W	FC 0x03, 0x06, 0x10

Example Usage

The address of interest is 0x30123. The relative address is 0x0123 and the only possible function is “0x04 Read Input register”.

If the address of interest is 0x40234, then possible function codes are 0x03, 0x06 and 0x10. Please see supported function codes in section “Supported Function Codes”.

6.2 Status Coil Registers

6.2.1 General Device Information

Reg. Addresses	Description	Type	R/W	Notes
0x00000	Selftest passed	bool	R	1 passed, 0 failed
0x00001	Device Ready	bool	R	1 Booting finished
0x00100	Synchronize Trigger	bool	W	Sync/Start trigger for all data sources

6.2.2 Acoustic Emission (AE) Sensor

Reg. Addresses	Description	Type	R/W	Notes
0x01000	Data Ready	bool	R	Auto resetting when read
0x01001	sPR Ref Values Ready	bool	R	
0x01100	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x01101	Use K(t) Ref Values	bool	W	See [COIL1]
0x01102	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x01103	Synchronize Trigger	bool	W	Sync/Start trigger for AE sensor

[COIL1] Copy K(t) RMS and Absolute Peak
Copy the current RMS and absolute peak runtime values into the K(t) RMS and absolute peak reference registers. Values are applied immediately and are only valid for current runtime. Set coil “Save Configuration” to make setting persistent.

6.2.3 Accelerometer

6.2.3.1 Accelerometer XYZ Channel

Reg. Addresses	Description	Type	R/W	Notes
0x02000	Data Ready	bool	R	Auto resetting when read
0x02001	sPR Ref Values Ready	bool	R	
0x02100	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x02101	Use K(t) Ref Values	bool	W	See [COIL1]
0x02102	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x02103	Synchronize Trigger	bool	W	Sync/Start trigger for accelerometer XYZ channel

6.2.3.2 Accelerometer X Channel

Reg. Addresses	Description	Type	R/W	Notes
0x02200	Data Ready	bool	R	Auto resetting when read
0x02201	sPR Ref Values Ready	bool	R	
0x02300	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x02301	Use K(t) Ref Values	bool	W	See [COIL1]
0x02302	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x02303	Synchronize Trigger	bool	W	Sync/Start trigger for accelerometer X channel

6.2.3.3 Accelerometer Y Channel

Reg. Addresses	Description	Type	R/W	Notes
0x02400	Data Ready	bool	R	Auto resetting when read
0x02401	sPR Ref Values Ready	bool	R	
0x02500	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x02501	Use K(t) Ref Values	bool	W	See [COIL1]
0x02502	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x02503	Synchronize Trigger	bool	W	Sync/Start trigger for accelerometer Y channel

6.2.3.4 Accelerometer Z Channel

Reg. Addresses	Description	Type	R/W	Notes
0x02600	Data Ready	bool	R	Auto resetting when read
0x02601	sPR Ref Values Ready	bool	R	
0x02700	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x02701	Use K(t) Ref Values	bool	W	See [COIL1]
0x02702	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x02703	Synchronize Trigger	bool	W	Sync/Start trigger for accelerometer Z channel

6.2.4 Gyroscope

6.2.4.1 Gyroscope XYZ Channel

Reg. Addresses	Description	Type	R/W	Notes
0x03000	Data Ready	bool	R	Auto resetting when read
0x03001	sPR Ref Values Ready	bool	R	
0x03100	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x03101	Use K(t) Ref Values	bool	W	See [COIL1]
0x03102	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x03103	Synchronize Trigger	bool	W	Sync/Start trigger for gyroscope XYZ channel

6.2.4.2 Gyroscope X Channel

Reg. Addresses	Description	Type	R/W	Notes
0x03200	Data Ready	bool	R	Auto resetting when read
0x03201	sPR Ref Values Ready	bool	R	
0x03300	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x03301	Use K(t) Ref Values	bool	W	See [COIL1]
0x03302	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x03303	Synchronize Trigger	bool	W	Sync/Start trigger for gyroscope X channel

6.2.4.3 Gyroscope Y Channel

Reg. Addresses	Description	Type	R/W	Notes
0x03400	Data Ready	bool	R	Auto resetting when read
0x03401	sPR Ref Values Ready	bool	R	
0x03500	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x03501	Use K(t) Ref Values	bool	W	See [COIL1]
0x03502	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x03503	Synchronize Trigger	bool	W	Sync/Start trigger for gyroscope Y channel

6.2.4.4 Gyroscope Z Channel

Reg. Addresses	Description	Type	R/W	Notes
0x03600	Data Ready	bool	R	Auto resetting when read
0x03601	sPR Ref Values Ready	bool	R	
0x03700	Update Dig. Filter	bool	R/W	Resets to ‘0’ after filter is calculated
0x03701	Use K(t) Ref Values	bool	W	See [COIL1]
0x03702	Use sPR Ref Values	bool	W	Use current measurement as sPR reference
0x03703	Synchronize Trigger	bool	W	Sync/Start trigger for gyroscope Z channel

6.2.5 Autonomous Condition Monitoring

Reg. Addresses	Description	Type	R/W	Notes
0x05000	CM 1 Alert	bool	R/W	See [COIL2]
0x05001	CM 2 Alert	bool	R/W	See [COIL2]
0x05002	CM 3 Alert	bool	R/W	See [COIL2]
0x05003	CM 4 Alert	bool	R/W	See [COIL2]
0x05004	CM 5 Alert	bool	R/W	See [COIL2]
0x05005	CM Logic A Alert	bool	R/W	See [COIL2]
0x05006	CM Logic B Alert	bool	R/W	See [COIL2]
0x05007	CM Logic C Alert	bool	R/W	See [COIL2]
0x05300	CM 1 First Alert	bool	R/W	See [COIL3]
0x05301	CM 2 First Alert	bool	R/W	See [COIL3]
0x05302	CM 3 First Alert	bool	R/W	See [COIL3]
0x05303	CM 4 First Alert	bool	R/W	See [COIL3]
0x05304	CM 5 First Alert	bool	R/W	See [COIL3]
0x05305	CM Logic A 1st Alert	bool	R/W	See [COIL3]
0x05306	CM Logic B 1st Alert	bool	R/W	See [COIL3]
0x05307	CM Logic C 1st Alert	bool	R/W	See [COIL3]

[COIL2] Condition Monitoring Alert
0 no alert, 1 alert active
Write ‘0’ to reset latched type alert

[COIL3] Condition Monitoring First Alert
0 first alert active, 1 first alert not triggered
Write ‘1’ to reset first alert flag.
This flag will change from ‘1’ to ‘0’ as soon as alert is triggered the first time and will indicate, that first alert value and first alert index are saved.
If first alert flag is reset, the first alert value and first alert index will also be reset.

6.2.6 Device Configuration

Reg. Addresses	Description	Type	R/W	Notes
0x0FFFD	Save Configuration	bool	R/W	1 save config, see [COIL4]
0x0FFFE	Factory Reset	bool	R/W	1 do factory reset, see [COIL5]
0x0FFFF	Reboot Device	bool	W	1 do reboot, see [COIL6]

[COIL4] Save Configuration
Make current configuration persistent. Coil will stay at ‘1’ as long as configuration is saved and resets to ‘0’ if configuration is saved.

[COIL5] Factory Reset
Coil will stay at ‘1’ as long as factory reset is done and resets to ‘0’ if factory reset is done. Set coil “Save Configuration” to make factory reset persistent.
Warning: All settings are reset. For example Modbus slave ID and speed will be reset immediately and device won’t respond if your current Modbus configuration is different.

[COIL6] Reboot Device
Reboot is done immediately, no response will be sent.

6.3 Measurement Registers

6.3.1 Data Type Tests

Reg. Addresses	Description	Type	R/W	Notes
0x30000	uint16 test	u16	R	returns 60123
0x30001	int16 test	i16	R	returns -30123
0x30002 - 0x30003	uint32 test	u32	R	returns 4000123456
0x30004 - 0x30005	float32 test	f32	R	returns 123.125

6.3.2 Temperature Sensors

Reg. Addresses	Description	Type	R/W	Notes
0x30010	Sensor Temperature	u16	R	see [MEA.TS1]
0x30011	CPU Temperature	u16	R	see [MEA.TS1]

[MEA.TS1] Unit Conversion of Temperature sensors
Read value is in Deci-Kelvin
Conversion: T = uint16_value / 10 - 273.15 (°C)

6.3.3 Electrical Conditions

Reg. Addresses	Description	Type	R/W	Notes
0x30020 - 0x30021	Input Voltage (V)	f32	R	available since fw v1.1.0
0x30022 - 0x30023	Input Current (A)	f32	R	available since fw v1.1.0

6.3.4 GPIO Channel

6.3.4.1 GPIO 1

Reg. Addresses	Description	Type	R/W	Notes
0x30030	Pulses per second	u16	R	updated each second
0x30031 - 0x30032	RPM rounds per minute	f32	R	updated each second

6.3.4.2 GPIO 2

Reg. Addresses	Description	Type	R/W	Notes
0x30040	Pulses per second	u16	R	updated each second
0x30041 - 0x30042	RPM rounds per minute	f32	R	updated each second

6.3.5 Acoustic Emission (AE) Sensor

6.3.5.1 Acoustic Emission (AE) Sensor

Reg. Addresses	Description	Type	R/W	Notes
0x31000 - 0x31001	Measurement Index	u32	R	Continuous Counter
0x31002 - 0x31003	Range	f32	R	24 bit resolution
0x31004 - 0x31005	Absolute Peak	f32	R	24 bit resolution
0x31006 - 0x31007	Mean	f32	R	24 bit resolution
0x31008 - 0x31009	RMS	f32	R	24 bit resolution
0x3100A - 0x3100B	Zero Crossing Rate	f32	R	24 bit resolution
0x3100C - 0x3100D	Crest Factor	f32	R	24 bit resolution
0x3100E - 0x3100F	K(t)	f32	R	24 bit resolution
0x31010 - 0x31011	SNR	f32	R	24 bit resolution
0x31012 - 0x31013	Skewness	f32	R	24 bit resolution
0x31014 - 0x31015	Kurtosis	f32	R	24 bit resolution
0x31016 - 0x31017	sPR parameter	f32	R	32 bit resolution
0x31018 - 0x31019	sPRs parameter	f32	R	32 bit resolution
0x3101A - 0x3101B	sPRc parameter	f32	R	32 bit resolution
0x3101C - 0x3101D	sPRsc parameter	f32	R	32 bit resolution
0x3101E - 0x3101F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.5.2 Acoustic Emission (AE) Sensor 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can't handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x31800 - 0x31801	Measurement Index	u32	R	Continuous Counter
0x31802	16-bit Range	u16	R	Fact: * 3 / 65535
0x31804	16-bit Absolute Peak	u16	R	Fact: * 1.5 / 65535
0x31806	16-bit Mean	i16	R	Fact: * 1.5 / 32767
0x31808	16-bit RMS	u16	R	Fact: * 1.5 / 32767
0x3180A	16-bit ZCR	u16	R	Fact: * 1
0x3180C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x3180E	16-bit K(t)	u16	R	Fact: * 1.5 / 65535
0x31810	16-bit SNR	i16	R	Fact: * 1 / 32767
0x31812	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x31814	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x31816	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x31818	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x3181A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x3181C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x3181E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.6 Accelerometer

6.3.6.1 Accelerometer XYZ Channel

Reg. Addresses	Description	Type	R/W	Notes
0x32000 - 0x32001	Measurement Index	u32	R	Continuous Counter
0x32002 - 0x32003	Range	f32	R	16 bit resolution
0x32004 - 0x32005	Absolute Peak	f32	R	16 bit resolution
0x32006 - 0x32007	Mean	f32	R	16 bit resolution
0x32008 - 0x32009	RMS	f32	R	16 bit resolution
0x3200A - 0x3200B	Zero Crossing Rate	f32	R	16 bit resolution
0x3200C - 0x3200D	Crest Factor	f32	R	16 bit resolution
0x3200E - 0x3200F	K(t)	f32	R	16 bit resolution
0x32010 - 0x32011	SNR	f32	R	16 bit resolution
0x32012 - 0x32013	Skewness	f32	R	16 bit resolution
0x32014 - 0x32015	Kurtosis	f32	R	16 bit resolution
0x32016 - 0x32017	sPR parameter	f32	R	32 bit resolution
0x32018 - 0x32019	sPRs parameter	f32	R	32 bit resolution
0x3201A - 0x3201B	sPRc parameter	f32	R	32 bit resolution
0x3201C - 0x3201D	sPRsc parameter	f32	R	32 bit resolution
0x3201E - 0x3201F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.6.2 Accelerometer X Channel

Reg. Addresses	Description	Type	R/W	Notes
0x32200 - 0x32201	Measurement Index	u32	R	Continuous Counter
0x32202 - 0x32203	Range	f32	R	16 bit resolution
0x32204 - 0x32205	Absolute Peak	f32	R	16 bit resolution
0x32206 - 0x32207	Mean	f32	R	16 bit resolution
0x32208 - 0x32209	RMS	f32	R	16 bit resolution
0x3220A - 0x3220B	Zero Crossing Rate	f32	R	16 bit resolution
0x3220C - 0x3220D	Crest Factor	f32	R	16 bit resolution
0x3220E - 0x3220F	K(t)	f32	R	16 bit resolution
0x32210 - 0x32211	SNR	f32	R	16 bit resolution
0x32212 - 0x32213	Skewness	f32	R	16 bit resolution
0x32214 - 0x32215	Kurtosis	f32	R	16 bit resolution
0x32216 - 0x32217	sPR parameter	f32	R	32 bit resolution
0x32218 - 0x32219	sPRs parameter	f32	R	32 bit resolution
0x3221A - 0x3221B	sPRc parameter	f32	R	32 bit resolution
0x3221C - 0x3221D	sPRsc parameter	f32	R	32 bit resolution
0x3221E - 0x3221F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.6.3 Accelerometer Y Channel

Reg. Addresses	Description	Type	R/W	Notes
0x32400 - 0x32401	Measurement Index	u32	R	Continuous Counter
0x32402 - 0x32403	Range	f32	R	16 bit resolution
0x32404 - 0x32405	Absolute Peak	f32	R	16 bit resolution
0x32406 - 0x32407	Mean	f32	R	16 bit resolution
0x32408 - 0x32409	RMS	f32	R	16 bit resolution
0x3240A - 0x3240B	Zero Crossing Rate	f32	R	16 bit resolution
0x3240C - 0x3240D	Crest Factor	f32	R	16 bit resolution
0x3240E - 0x3240F	K(t)	f32	R	16 bit resolution
0x32410 - 0x32411	SNR	f32	R	16 bit resolution
0x32412 - 0x32413	Skewness	f32	R	16 bit resolution
0x32414 - 0x32415	Kurtosis	f32	R	16 bit resolution
0x32416 - 0x32417	sPR parameter	f32	R	32 bit resolution
0x32418 - 0x32419	sPRs parameter	f32	R	32 bit resolution
0x3241A - 0x3241B	sPRc parameter	f32	R	32 bit resolution
0x3241C - 0x3241D	sPRsc parameter	f32	R	32 bit resolution
0x3241E - 0x3241F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.6.4 Accelerometer Z Channel

Reg. Addresses	Description	Type	R/W	Notes
0x32600 - 0x32601	Measurement Index	u32	R	Continuous Counter
0x32602 - 0x32603	Range	f32	R	16 bit resolution
0x32604 - 0x32605	Absolute Peak	f32	R	16 bit resolution
0x32606 - 0x32607	Mean	f32	R	16 bit resolution
0x32608 - 0x32609	RMS	f32	R	16 bit resolution
0x3260A - 0x3260B	Zero Crossing Rate	f32	R	16 bit resolution
0x3260C - 0x3260D	Crest Factor	f32	R	16 bit resolution
0x3260E - 0x3260F	K(t)	f32	R	16 bit resolution
0x32610 - 0x32611	SNR	f32	R	16 bit resolution
0x32612 - 0x32613	Skewness	f32	R	16 bit resolution
0x32614 - 0x32615	Kurtosis	f32	R	16 bit resolution
0x32616 - 0x32617	sPR parameter	f32	R	32 bit resolution
0x32618 - 0x32619	sPRs parameter	f32	R	32 bit resolution
0x3261A - 0x3261B	sPRc parameter	f32	R	32 bit resolution
0x3261C - 0x3261D	sPRsc parameter	f32	R	32 bit resolution
0x3261E - 0x3261F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.6.5 Accelerometer XYZ Channel 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can't handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x32800 - 0x32801	Measurement Index	u32	R	Continuous Counter
0x32802	16-bit Range	u16	R	Fact: * 32 / 65535
0x32804	16-bit Absolute Peak	u16	R	Fact: * 16 / 65535
0x32806	16-bit Mean	i16	R	Fact: * 16 / 32767
0x32808	16-bit RMS	u16	R	Fact: * 16 / 65535
0x3280A	16-bit ZCR	u16	R	Fact: * 1
0x3280C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x3280E	16-bit K(t)	u16	R	Fact: * 16 / 65535
0x32810	16-bit SNR	i16	R	Fact: * 1 / 32767
0x32812	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x32814	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x32816	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x32818	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x3281A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x3281C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x3281E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.6.6 Accelerometer X Channel 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can't handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x32A00 - 0x32A01	Measurement Index	u32	R	Continuous Counter
0x32A02	16-bit Range	u16	R	Fact: * 32 / 65535
0x32A04	16-bit Absolute Peak	u16	R	Fact: * 16 / 65535
0x32A06	16-bit Mean	i16	R	Fact: * 16 / 32767
0x32A08	16-bit RMS	u16	R	Fact: * 16 / 65535
0x32A0A	16-bit ZCR	u16	R	Fact: * 1
0x32A0C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x32A0E	16-bit K(t)	u16	R	Fact: * 16 / 65535
0x32A10	16-bit SNR	i16	R	Fact: * 1 / 32767
0x32A12	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x32A14	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x32A16	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x32A18	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x32A1A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x32A1C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x32A1E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.6.7 Accelerometer Y Channel 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can’t handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x32C00 - 0x32C01	Measurement Index	u32	R	Continuous Counter
0x32C02	16-bit Range	u16	R	Fact: * 32 / 65535
0x32C04	16-bit Absolute Peak	u16	R	Fact: * 16 / 65535
0x32C06	16-bit Mean	i16	R	Fact: * 16 / 32767
0x32C08	16-bit RMS	u16	R	Fact: * 16 / 65535
0x32C0A	16-bit ZCR	u16	R	Fact: * 1
0x32C0C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x32C0E	16-bit K(t)	u16	R	Fact: * 16 / 65535
0x32C10	16-bit SNR	i16	R	Fact: * 1 / 32767
0x32C12	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x32C14	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x32C16	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x32C18	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x32C1A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x32C1C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x32C1E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.6.8 Accelerometer Z Channel 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can't handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x32E00 - 0x32E01	Measurement Index	u32	R	Continuous Counter
0x32E02	16-bit Range	u16	R	Fact: * 32 / 65535
0x32E04	16-bit Absolute Peak	u16	R	Fact: * 16 / 65535
0x32E06	16-bit Mean	i16	R	Fact: * 16 / 32767
0x32E08	16-bit RMS	u16	R	Fact: * 16 / 65535
0x32E0A	16-bit ZCR	u16	R	Fact: * 1
0x32E0C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x32E0E	16-bit K(t)	u16	R	Fact: * 16 / 65535
0x32E10	16-bit SNR	i16	R	Fact: * 1 / 32767
0x32E12	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x32E14	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x32E16	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x32E18	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x32E1A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x32E1C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x32E1E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.7 Gyroscope

6.3.7.1 Gyroscope XYZ Channel

Reg. Addresses	Description	Type	R/W	Notes
0x33000 - 0x33001	Measurement Index	u32	R	Continuous Counter
0x33002 - 0x33003	Range	f32	R	16 bit resolution
0x33004 - 0x33005	Absolute Peak	f32	R	16 bit resolution
0x33006 - 0x33007	Mean	f32	R	16 bit resolution
0x33008 - 0x33009	RMS	f32	R	16 bit resolution
0x3300A - 0x3300B	Zero Crossing Rate	f32	R	16 bit resolution
0x3300C - 0x3300D	Crest Factor	f32	R	16 bit resolution
0x3300E - 0x3300F	K(t)	f32	R	16 bit resolution
0x33010 - 0x33011	SNR	f32	R	16 bit resolution
0x33012 - 0x33013	Skewness	f32	R	16 bit resolution
0x33014 - 0x33015	Kurtosis	f32	R	16 bit resolution
0x33016 - 0x33017	sPR parameter	f32	R	32 bit resolution
0x33018 - 0x33019	sPRs parameter	f32	R	32 bit resolution
0x3301A - 0x3301B	sPRc parameter	f32	R	32 bit resolution
0x3301C - 0x3301D	sPRsc parameter	f32	R	32 bit resolution
0x3301E - 0x3301F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.7.2 Gyroscope X Channel

Reg. Addresses	Description	Type	R/W	Notes
0x33200 - 0x33201	Measurement Index	u32	R	Continuous Counter
0x33202 - 0x33203	Range	f32	R	16 bit resolution
0x33204 - 0x33205	Absolute Peak	f32	R	16 bit resolution
0x33206 - 0x33207	Mean	f32	R	16 bit resolution
0x33208 - 0x33209	RMS	f32	R	16 bit resolution
0x3320A - 0x3320B	Zero Crossing Rate	f32	R	16 bit resolution
0x3320C - 0x3320D	Crest Factor	f32	R	16 bit resolution
0x3320E - 0x3320F	K(t)	f32	R	16 bit resolution
0x33210 - 0x33211	SNR	f32	R	16 bit resolution
0x33212 - 0x33213	Skewness	f32	R	16 bit resolution
0x33214 - 0x33215	Kurtosis	f32	R	16 bit resolution
0x33216 - 0x33217	sPR parameter	f32	R	32 bit resolution
0x33218 - 0x33219	sPRs parameter	f32	R	32 bit resolution
0x3321A - 0x3321B	sPRc parameter	f32	R	32 bit resolution
0x3321C - 0x3321D	sPRsc parameter	f32	R	32 bit resolution
0x3321E - 0x3321F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.7.3 Gyroscope Y Channel

Reg. Addresses	Description	Type	R/W	Notes
0x33400 - 0x33401	Measurement Index	u32	R	Continuous Counter
0x33402 - 0x33403	Range	f32	R	16 bit resolution
0x33404 - 0x33405	Absolute Peak	f32	R	16 bit resolution
0x33406 - 0x33407	Mean	f32	R	16 bit resolution
0x33408 - 0x33409	RMS	f32	R	16 bit resolution
0x3340A - 0x3340B	Zero Crossing Rate	f32	R	16 bit resolution
0x3340C - 0x3340D	Crest Factor	f32	R	16 bit resolution
0x3340E - 0x3340F	K(t)	f32	R	16 bit resolution
0x33410 - 0x33411	SNR	f32	R	16 bit resolution
0x33412 - 0x33413	Skewness	f32	R	16 bit resolution
0x33414 - 0x33415	Kurtosis	f32	R	16 bit resolution
0x33416 - 0x33417	sPR parameter	f32	R	32 bit resolution
0x33418 - 0x33419	sPRs parameter	f32	R	32 bit resolution
0x3341A - 0x3341B	sPRc parameter	f32	R	32 bit resolution
0x3341C - 0x3341D	sPRsc parameter	f32	R	32 bit resolution
0x3341E - 0x3341F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.7.4 Gyroscope Z Channel

Reg. Addresses	Description	Type	R/W	Notes
0x33600 - 0x33601	Measurement Index	u32	R	Continuous Counter
0x33602 - 0x33603	Range	f32	R	16 bit resolution
0x33604 - 0x33605	Absolute Peak	f32	R	16 bit resolution
0x33606 - 0x33607	Mean	f32	R	16 bit resolution
0x33608 - 0x33609	RMS	f32	R	16 bit resolution
0x3360A - 0x3360B	Zero Crossing Rate	f32	R	16 bit resolution
0x3360C - 0x3360D	Crest Factor	f32	R	16 bit resolution
0x3360E - 0x3360F	K(t)	f32	R	16 bit resolution
0x33610 - 0x33611	SNR	f32	R	16 bit resolution
0x33612 - 0x33613	Skewness	f32	R	16 bit resolution
0x33614 - 0x33615	Kurtosis	f32	R	16 bit resolution
0x33616 - 0x33617	sPR parameter	f32	R	32 bit resolution
0x33618 - 0x33619	sPRs parameter	f32	R	32 bit resolution
0x3361A - 0x3361B	sPRc parameter	f32	R	32 bit resolution
0x3361C - 0x3361D	sPRsc parameter	f32	R	32 bit resolution
0x3361E - 0x3361F	Elapsed Win Len (s)	f32	R	Length of previous calculation Window

6.3.7.5 Gyroscope XYZ Channel 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can't handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x33800 - 0x33801	Measurement Index	u32	R	Continuous Counter
0x33802	16-bit Range	u16	R	Fact: * 4000 / 65535
0x33804	16-bit Absolute Peak	u16	R	Fact: * 2000 / 65535
0x33806	16-bit Mean	i16	R	Fact: * 2000 / 32767
0x33808	16-bit RMS	u16	R	Fact: * 2000 / 65535
0x3380A	16-bit ZCR	u16	R	Fact: * 1
0x3380C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x3380E	16-bit K(t)	u16	R	Fact: * 2000 / 65535
0x33810	16-bit SNR	i16	R	Fact: * 1 / 32767
0x33812	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x33814	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x33816	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x33818	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x3381A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x3381C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x3381E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.7.6 Gyroscope X Channel 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can't handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x33A00 - 0x33A01	Measurement Index	u32	R	Continuous Counter
0x33A02	16-bit Range	u16	R	Fact: * 4000 / 65535
0x33A04	16-bit Absolute Peak	u16	R	Fact: * 2000 / 65535
0x33A06	16-bit Mean	i16	R	Fact: * 2000 / 32767
0x33A08	16-bit RMS	u16	R	Fact: * 2000 / 65535
0x33A0A	16-bit ZCR	u16	R	Fact: * 1
0x33A0C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x33A0E	16-bit K(t)	u16	R	Fact: * 2000 / 65535
0x33A10	16-bit SNR	i16	R	Fact: * 1 / 32767
0x33A12	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x33A14	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x33A16	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x33A18	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x33A1A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x33A1C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x33A1E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.7.7 Gyroscope Y Channel 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can't handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x33C00 - 0x33C01	Measurement Index	u32	R	Continuous Counter
0x33C02	16-bit Range	u16	R	Fact: * 4000 / 65535
0x33C04	16-bit Absolute Peak	u16	R	Fact: * 2000 / 65535
0x33C06	16-bit Mean	i16	R	Fact: * 2000 / 32767
0x33C08	16-bit RMS	u16	R	Fact: * 2000 / 65535
0x33C0A	16-bit ZCR	u16	R	Fact: * 1
0x33C0C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x33C0E	16-bit K(t)	u16	R	Fact: * 2000 / 65535
0x33C10	16-bit SNR	i16	R	Fact: * 1 / 32767
0x33C12	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x33C14	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x33C16	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x33C18	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x33C1A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x33C1C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x33C1E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.7.8 Gyroscope Z Channel 16-bit (legacy) values

16-bit (legacy) values are provided for systems which can’t handle float32 values.

Registers between two 16-bit values read as 0x00. A factor to calculate the related float32 register value is provided for each 16-bit register.

Reg. Addresses	Description	Type	R/W	Notes
0x33E00 - 0x33E01	Measurement Index	u32	R	Continuous Counter
0x33E02	16-bit Range	u16	R	Fact: * 4000 / 65535
0x33E04	16-bit Absolute Peak	u16	R	Fact: * 2000 / 65535
0x33E06	16-bit Mean	i16	R	Fact: * 2000 / 32767
0x33E08	16-bit RMS	u16	R	Fact: * 2000 / 65535
0x33E0A	16-bit ZCR	u16	R	Fact: * 1
0x33E0C	16-bit Crest Factor	u16	R	Fact: * 1 / 100
0x33E0E	16-bit K(t)	u16	R	Fact: * 2000 / 65535
0x33E10	16-bit SNR	i16	R	Fact: * 1 / 32767
0x33E12	16-bit Skewness	i16	R	Fact: * 10 / 32767
0x33E14	16-bit Kurtosis	i16	R	Fact: * 10 / 32767
0x33E16	16-bit sPR parameter	u16	R	Fact: * 1 / 100
0x33E18	16-bit sPRs parameter	u16	R	Fact: * 1 / 100
0x33E1A	16-bit sPRc parameter	u16	R	Fact: * 1 / 100
0x33E1C	16-bit sPRsc param	u16	R	Fact: * 1 / 100
0x33E1E	16-bit Elapse Win (s)	u16	R	Fact: * 1 / 100

6.3.8 Data Recorder

The recorded data is saved in the smartPREDICT's SDRAM.
Access to the data is granted by the register addresses 0x3400A – 0x34085 which represent 62 Samples of data. The index of Data0 is provided over register address 0x34008 – 0x34009. Index n of Data[n] is the offset to the Index of Data0. The index of Data0 is incremented every time, the MSB (high) word register of Data[n] is read. The content of Data[n] is updated each time, the MSB (high) word register of Data0 is read.

If data is lost, the index of Data0 can be set by the configuration register addresses 0x44005 – 0x44006.

If multiple registers of data are read via Modbus command "0x04 Read Input Registers", Data0 register must be included, so that index of Data0 register matches to the read data samples.

Reg. Addresses	Description	Type	R/W	Notes
0x34000 - 0x34001	Time remaining	f32	R	Till end of recording (s)
0x34002 - 0x34003	Samples remaining	u32	R	Till end of recording
0x34004 - 0x34005	Time recorded	f32	R	Since start of recording (s)
0x34006 - 0x34007	Samples recorded	u32	R	Since start of recording
0x34008 - 0x34009	Index of Data0	u32	R	
0x3400A - 0x3400B	Data0	f32	R	
0x3400C - 0x3400D	Data1	f32	R	
0x3400E - 0x3400F	Data2	f32	R	
0x34010 - 0x34011	Data3	f32	R	
0x34012 - 0x34013	Data4	f32	R	
0x34014 - 0x34015	Data5	f32	R	
0x34016 - 0x34017	Data6	f32	R	
0x34018 - 0x34019	Data7	f32	R	
0x3401A - 0x3401C	Data8	f32	R	
0x3401C - 0x3401E	Data9	f32	R	
0x3401E - 0x3401F	Data10	f32	R	
…	Data[n]	f32	R	
0x34082 - 0x34083	Data60	f32	R	
0x34084 - 0x34085	Data61	f32	R	
0x34086 - 0x34087	PSD df	f32	R	frequency-resolution (only PSD)
0x34088 - 0x34089	PSD dn	f32	R	dn = df / RPM (1/s) (only PSD)

6.4 Configuration Registers

If not otherwise stated, all configurations are only valid during current runtime. Write coil “Save Configuration” to make setting persistent.
If settings are more than one register wide, changes to runtime are applied after LSB (low) word is received.

6.4.1 Device Information

Reg. Addresses	Description	Type	R/W	Notes
0x40000 - 0x40001	Unique Hardware ID	u32	R	
0x40002	Hardware Version	u16	R	
0x40003 - 0x40004	Firmware Version	u32	R	
0x40005	USER LED Control	u16	R/W	see [CFG.DEV1]
0x40006 - 0x40007	Float32 write test	f32	R/W	For loopback tests
0x40008 - 0x40009	Selftest Result	u32	R	
0x4000A	Main sensor HW	u16	R	0 chAMP, 1 ADXL100x
0x4000B	Acc/Gyro sensor HW	u16	R	0 Bosch BMI160 1 STMicroelectronics LSM6DSO32X 2 STMicroelectronics LSM6DSO32X and ANALOG DEVICES ADXL373
0x4000F	Active application	u16	R	return always 0x8004

[CFG.DEV1] USER LED Control
0 off, 1 blue, 2 green, 3 cyan, 4 red, 5 magenta, 6 yellow, 7 white

6.4.2 Communication

Reg. Addresses	Description	Type	R/W	Notes
0x40010	Operation Mode	u16	R/W	0 Modbus
0x40011	Modbus Slave Address	u16	R/W	1…247, 1 default
0x40012	Modbus Mode	u16	R/W	0 RTU/Binary
0x40013	Modbus Speed	u16	R/W	see [CFG.COM1]
0x40014	Mb Term. Resistor	u16	R/W	see [CFG.COM2]

[CFG.COM1] Modbus Speed
2 4800, 3 9600, 4 19200 (default), 5 38400, 6 56000, 7 57600, 8 115200, 9 128000, 10 230400, 11 256000, 12 460800, 13 500000, 14 576000, 15 921600, 16 1000000, 17 1500000, 18 2000000, 19 3000000, 20 6000000, 21 12000000, 22 13500000
Configuration will be applied immediately. A response will be sent with the new Modbus speed. Save request is required to make configuration persistent.
It’s only possible to save Modbus speed values up to 115200 persistent.
Hardware Version 4 only supports Modbus speed values up to 500000

[CFG.COM2] Modbus Termination Resistor Setting
0 Off (default), 1 on
Setting is available in Hardware Version 4 and later

6.4.3 GPIO Channel

6.4.3.1 GPIO 1

Reg. Addresses	Description	Type	R/W	Notes
0x40020	GPIO Mode	u16	R/W	See [CFG.GPIO1]
0x40021	Pull- / Pull-Down	u16	R/W	See [CFG.GPIO2]
0x40022	High- / Low-Active	u16	R/W	See [CFG.GPIO3]
0x40023	Pulses per round	u16	R/W	See [CFG.GPIO4]

[CFG.GPIO1] GPIO Mode
0 Off, 1 Autonomous Condition Monitor (default), 2 Trigger Measurement Start / Stop, 4 Pulse counter per second
Mode 4 “Pulse counter per second” is only available for GPIO 1 at the moment

[CFG.GPIO2] GPIO Pull-Up / Pull-Down Setting
0 No-Pull, 1 Pull-Up, 2 Pull-Down (default)
Not available in Modes 0 “Off” and 1 “Autonomous Condition Monitor”
In GPIO mode 2 “Trigger Measurement Start / Stop” only setting 2 “Pull-Down” is available
In GPIO mode 4 “Pulse counter per second” all settings are available

[CFG.GPIO3] GPIO High-Active / Low-Active
0 High-Active
Only available in GPIO Mode 2 “Trigger Measurement Start / Stop”

[CFG.GPIO4] GPIO Pulses per round
1 (default), range: 1…65535
Only available in GPIO mode 4 “Pulse counter per second”

6.4.3.2 GPIO 2

Reg. Addresses	Description	Type	R/W	Notes
0x40030	GPIO Mode	u16	R/W	See [CFG.GPIO1]
0x40031	Pull- / Pull-Down	u16	R/W	See [CFG.GPIO2]
0x40032	High- / Low-Active	u16	R/W	See [CFG.GPIO3]
0x40033	Pulses per round	u16	R/W	See [CFG.GPIO4]

6.4.4 Acoustic Emission (AE) Sensor

6.4.4.1 Acoustic Emission Sensor

Reg. Addresses	Description	Type	R/W	Notes
0x41000	Mode	u16	R/W	See [CFG.AE1]
0x41001	Sample Rate	u16	R/W	See [CFG.AE2]
0x41002	Analog filter freq.	u16	R/W	See [CFG.AE3]
0x41003	Analog filter gain	u16	R/W	See [CFG.AE4]
0x41004	Filter Type	u16	R/W	See [CFG.IIR1]
0x41005	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x41006	Filter Order	u16	R/W	See [CFG.IIR3]
0x41007 - 0x41008	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x41009 - 0x4100A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4100B - 0x4100C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4100D - 0x4100E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4100F - 0x41010	Window Length (s)	f32	R/W	Default: 1.0 (s), Min: 0.128
0x41011 - 0x41012	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x41013 - 0x41014	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x41015 - 0x41016	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x41017 - 0x41018	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x41019 - 0x4101A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4101B - 0x4101C	sPR Tuning High	f32	R/W	default: 48000.0 (Hz)
0x4101D - 0x4101E	sPRc Tuning Low	f32	R/W	default: 0.0 [Hz]
0x4101F - 0x41020	sPRc Tuning High	f32	R/W	default: 48000.0 [Hz]

[CFG.AE1] AE Mode
0 Off, 1 Continuous Window Length (default), 4 Continuous GPIO Triggered Start / Stop

[CFG.AE2] AE Sample Rate
3 8 kHz, 4 16 kHz, 5 32 kHz, 6 48 kHz, 7 64 kHz, 8 96 kHz (default)

[CFG.AE3] AE analog filter frequency
1 Low, 2 Medium (default), 3 High - see AE gain vs. lower cut-off frequency table
Don’t care for ADXL100x (smartPREDICT-ID RS485/USB)

[CFG.AE4] AE analog filter gain
0 -20 dB, 1 0 dB (default), 2 +20 dB, 3 Max - see AE gain vs. lower cut-off frequency table
Don’t care for ADXL100x (smartPREDICT-ID RS485/USB)

[CFG.IIR1] Digital IIR Signal Filter: Filter Type (Topology)
0 Off (default), 1 Butterworth, 2 Chebyshev, 4 Elliptic, 5 Bessel
3 Inverse Chebyshev will be added with a future update
Configuration is applied after writing coil “Update Dig. Filter”

[CFG.IIR2] Digital IIR Signal Filter: Pass Type
0 Lowpass (default), 1 Highpass, 3 Bandpass, 4 Notch
Configuration is applied after writing coil “Update Dig. Filter”

[CFG.IIR3] Digital IIR Signal Filter: Order
1…12, default: 1
Configuration is applied after writing coil “Update Dig. Filter”

[CFG.IIR4] Digital IIR Signal Filter: Lower corner (center) frequency
default: 100.0 (Hz)
Center frequency for lowpass and highpass, lower corner frequency for bandpass and notch
Configuration is applied after writing coil “Update Dig. Filter”

[CFG.IIR5] Digital IIR Signal Filter: Upper corner frequency
default: 200.0 (Hz)
Upper corner frequency for bandpass and notch
Configuration is applied after writing coil “Update Dig. Filter”

[CFG.IIR6] Digital IIR Signal Filter: Passband Ripple
0.0001 <= Ripple <= 1.0 dB, default: 1.0
Passband ripple for Chebyshev and Elliptic
Configuration is applied after writing coil “Update Dig. Filter”

[CFG.IIR7] Digital IIR Signal Filter: Stopband Attenuation
20 <= StopBand <= 120 dB, default: 60.0
Stopband attenuation for Inverse Chebyshev and Elliptic
Configuration is applied after writing coil “Update Dig. Filter”

[CFG.SPR1] sPR Reference Sample Rate
Configured sample rate (of data source), when sPR reference was set
Default: 0 (if no sPR reference is set)
If current sample rate is different to sample rate of reference, there is a chance of sPR can’t be calculated. In this case coil “sPR Ref Values Ready” if reset to ‘0’ and sPR parameter is not calculated.

[CFG.SPR2] sPR Reference Window Length
Configured window length (of data source), when sPR reference was set
Default: NAN (if no sPR reference is set)
If current window length is different to window length of reference, there is a chance of sPR can’t be calculated. In this case coil “sPR Ref Values Ready” if reset to ‘0’ and sPR parameter is not calculated.

6.4.4.2 AE Sensor chAMP Signal Generator

Reg. Addresses	Description	Type	R/W	Notes
0x41100	Signal Type	u16	R/W	See [CFG.AE4]
0x41101 - 0x41102	Signal Frequency	f32	R/W	See [CFG.AE5]

[CFG.AE4] AE Sensor chAMP Signal Generator: Signal Type
0 Off (default), 1 Sine, 2 Triangle, 3 Square, 4 Ramp
Don’t care for ADXL100x (smartPREDICT-ID RS485/USB)

[CFG.AE5] AE Sensor chAMP Signal Generator: Signal Frequency
Default value is 1000.0 (Hz). Value must be between 0.0 Hz and 423952.0 Hz
Note: Maximum sample rate of AE sensor is 96 kHz, so it’s not recommended to set at “Signal Frequency” above 48000 Hz
Don’t care for ADXL100x (smartPREDICT-ID RS485/USB)

6.4.5 Accelerometer

6.4.5.1 Accelerometer XYZ Channel

Reg. Addresses	Description	Type	R/W	Notes
0x42000	Mode	u16	R/W	See [CFG.ACC1]
0x42001	Sample Rate	u16	R/W	See [CFG.ACC2]
0x42002	Range	u16	R/W	See [CFG.ACC3]
0x42003	Reserved	u16	R/W	Read as ‘0’
0x42004	Filter Type	u16	R/W	See [CFG.IIR1]
0x42005	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x42006	Filter Order	u16	R/W	See [CFG.IIR3]
0x42007 - 0x42008	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x42009 - 0x4200A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4200B - 0x4200C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4200D - 0x4200E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4200F - 0x42010	Window Length (s)	f32	R/W	Default: 1.0 (s)
0x42011 - 0x42012	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x42013 - 0x42014	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x42015 - 0x42016	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x42017 - 0x42018	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x42019 - 0x4201A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4201B - 0x4201C	sPR Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)
0x4201D - 0x4201E	sPRc Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4201F - 0x42020	sPRc Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)

[CFG.ACC1] Accelerometer Mode
0 Off, 1 Continuous Window Length (default), 4 Continuous GPIO Triggered Start / Stop
Changing the mode of one accelerometer channel (XYZ, X, Y, Z) will change the mode of all accelerometer channel

[CFG.ACC2] Accelerometer Sample Rate
7 1600 (Hz) (default for BMI160), 17 6667 Hz (default for LSM6DSO32X and ADXL373)

[CFG.ACC3] Accelerometer Range
0 +/-2g (default BMI160), 1 +/-4g (default LSM6DSO32X), 2 +/-8g, 3 +/-16g, 4 +/-32g, 18 +/-400g
+/-2g is only available for BMI160; +/-32g is only available for LSM6DSO32X; +/-400g is only available for ADXL373
Changing the range of one accelerometer channel (XYZ, X, Y, Z) will change the range of all accelerometer

channel

6.4.5.2 Accelerometer X Channel

Reg. Addresses	Description	Type	R/W	Notes
0x42200	Mode	u16	R/W	See [CFG.ACC1]
0x42201	Sample Rate	u16	R/W	See [CFG.ACC2]
0x42202	Range	u16	R/W	See [CFG.ACC3]
0x42203	Reserved	u16	R/W	Read as '0'
0x42204	Filter Type	u16	R/W	See [CFG.IIR1]
0x42205	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x42206	Filter Order	u16	R/W	See [CFG.IIR3]
0x42207 - 0x42208	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x42209 -0 x4220A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4220B - 0x4220C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4220D - 0x4220E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4220F - 0x42210	Window Length (s)	f32	R/W	Default: 1.0 (s)
0x42211 - 0x42212	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x42213 - 0x42214	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x42215 - 0x42216	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x42217 - 0x42218	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x42219 - 0x4221A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4221B - 0x4221C	sPR Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)
0x4221D - 0x4221E	sPRc Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4221F - 0x42220	sPRc Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)

6.4.5.3 Accelerometer Y Channel

Reg. Addresses	Description	Type	R/W	Notes
0x42400	Mode	u16	R/W	See [CFG.ACC1]
0x42401	Sample Rate	u16	R/W	See [CFG.ACC2]
0x42402	Range	u16	R/W	See [CFG.ACC3]
0x42403	Reserved	u16	R/W	Read as ‘0’
0x42404	Filter Type	u16	R/W	See [CFG.IIR1]
0x42405	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x42406	Filter Order	u16	R/W	See [CFG.IIR3]
0x42407 - 0x42408	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x42409 - 0x4240A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4240B - 0x4240C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4240D - 0x4240E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4240F - 0x42410	Window Length (s)	f32	R/W	Default: 1.0 (s)
0x42411 - 0x42412	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x42413 - 0x42414	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x42415 - 0x42416	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x42417 - 0x42418	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x42419 - 0x4241A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4241B - 0x4241C	sPR Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)
0x4241D - 0x4241E	sPRc Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4241F - 0x42420	sPRc Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)

6.4.5.4 Accelerometer Z Channel

Reg. Addresses	Description	Type	R/W	Notes
0x42600	Mode	u16	R/W	See [CFG.ACC1]
0x42601	Sample Rate	u16	R/W	See [CFG.ACC2]
0x42602	Range	u16	R/W	See [CFG.ACC3]
0x42603	Reserved	u16	R/W	Read as ‘0’
0x42604	Filter Type	u16	R/W	See [CFG.IIR1]
0x42605	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x42606	Filter Order	u16	R/W	See [CFG.IIR3]
0x42607 - 0x42608	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x42609 - 0x4260A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4260B - 0x4260C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4260D - 0x4260E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4260F - 0x42610	Window Length (s)	f32	R/W	Default: 1.0 (s)
0x42611 - 0x42612	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x42613 - 0x42614	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x42615 - 0x42616	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x42617 - 0x42618	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x42619 - 0x4261A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4261B - 0x4261C	sPR Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)
0x4261D - 0x4261E	sPRc Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4261F - 0x42620	sPRc Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)

6.4.6 Gyroscope

6.4.6.1 Gyroscope XYZ Channel

Reg. Addresses	Description	Type	R/W	Notes
0x43000	Mode	u16	R/W	See [CFG.GYRO1]
0x43001	Sample Rate	u16	R/W	See [CFG.GYRO2]
0x43002	Range	u16	R/W	See [CFG.GYRO3]
0x43003	Reserved	u16	R/W	Read as ‘0’
0x43004	Filter Type	u16	R/W	See [CFG.IIR1]
0x43005	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x43006	Filter Order	u16	R/W	See [CFG.IIR3]
0x43007 - 0x43008	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x43009 - 0x4300A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4300B - 0x4300C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4300D - 0x4300E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4300F - 0x43010	Window Length (s)	f32	R/W	Default: 1.0 (s)
0x43011 - 0x43012	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x43013 - 0x43014	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x43015 - 0x43016	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x43017 - 0x43018	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x43019 - 0x4301A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4301B - 0x4301C	sPR Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)
0x4301D - 0x4301E	sPRc Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4301F - 0x43020	sPRc Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)

[CFG.GYRO1] Gyroscope Mode
0 Off, 1 Continuous Window Length (default), 4 Continuous GPIO Triggered Start / Stop
Changing the mode of one gyroscope channel (XYZ, X, Y, Z) will change the mode of all gyroscope channel

[CFG.GYRO2] Gyroscope Sample Rate
6 1600 (Hz) (default for BMI160), 17 6667 (Hz) (default for LSM6DSO32X)

[CFG.GYRO3] Gyroscope Range
0 +/-125 °/s (default), 1 +/-250 °/s, 2 +/-500 °/s, 3 +/-1000 °/s, 4 +/-2000 °/s
Changing the range of one gyroscope channel (XYZ, X, Y, Z) will change the range of all gyroscope channel

6.4.6.2 Gyroscope X Channel

Reg. Addresses	Description	Type	R/W	Notes
0x43200	Mode	u16	R/W	See [CFG.GYRO1]
0x43201	Sample Rate	u16	R/W	See [CFG.GYRO2]
0x43202	Range	u16	R/W	See [CFG.GYRO3]
0x43203	Reserved	u16	R/W	Read as ‘0’
0x43204	Filter Type	u16	R/W	See [CFG.IIR1]
0x43205	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x43206	Filter Order	u16	R/W	See [CFG.IIR3]
0x43207 - 0x43208	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x43209 - 0x4320A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4320B - 0x4320C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4320D - 0x4320E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4320F - 0x43210	Window Length (s)	f32	R/W	Default: 1.0 (s)
0x43211 - 0x43212	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x43213 - 0x43214	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x43215 - 0x43216	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x43217 - 0x43218	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x43219 - 0x4321A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4321B - 0x4321C	sPR Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)
0x4321D - 0x4321E	sPRc Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4321F - 0x43220	sPRc Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)

6.4.6.3 Gyroscope Y Channel

Reg. Addresses	Description	Type	R/W	Notes
0x43400	Mode	u16	R/W	See [CFG.GYRO1]
0x43401	Sample Rate	u16	R/W	See [CFG.GYRO2]
0x43402	Range	u16	R/W	See [CFG.GYRO3]
0x43403	Reserved	u16	R/W	Read as ‘0’
0x43404	Filter Type	u16	R/W	See [CFG.IIR1]
0x43405	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x43406	Filter Order	u16	R/W	See [CFG.IIR3]
0x43407 - 0x43408	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x43409 - 0x4340A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4340B - 0x4340C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4340D - 0x4340E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4340F - 0x43410	Window Length (s)	f32	R/W	Default: 1.0 (s)
0x43411 - 0x43412	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x43413 - 0x43414	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x43415 - 0x43416	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x43417 - 0x43418	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x43419 - 0x4341A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4341B - 0x4341C	sPR Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)
0x4341D - 0x4341E	sPRc Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4341F - 0x43420	sPRc Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)

6.4.6.4 Gyroscope Z Channel

Reg. Addresses	Description	Type	R/W	Notes
0x43600	Mode	u16	R/W	See [CFG.GYRO1]
0x43601	Sample Rate	u16	R/W	See [CFG.GYRO2]
0x43602	Range	u16	R/W	See [CFG.GYRO3]
0x43603	Reserved	u16	R/W	Read as '0'
0x43604	Filter Type	u16	R/W	See [CFG.IIR1]
0x43605	Filter Pass Type	u16	R/W	See [CFG.IIR2]
0x43606	Filter Order	u16	R/W	See [CFG.IIR3]
0x43607 - 0x43608	Filter Lower Fc	f32	R/W	See [CFG.IIR4]
0x43609 - 0x4360A	Filter Upper Fc	f32	R/W	See [CFG.IIR5]
0x4360B - 0x4360C	Filter PB Ripple	f32	R/W	See [CFG.IIR6]
0x4360D - 0x4360E	Filter SB Attenuation	f32	R/W	See [CFG.IIR7]
0x4360F - 0x43610	Window Length (s)	f32	R/W	Default: 1.0 (s)
0x43611 - 0x43612	K(t) Ref. Abs. Peak	f32	R/W	Reference for K(t) calculation
0x43613 - 0x43614	K(t) Ref. RMS	f32	R/W	Reference for K(t) calculation
0x43615 - 0x43616	sPR Ref. Sample Rate	u32	R	See [CFG.SPR1]
0x43617 - 0x43618	sPR Ref. Window Len.	f32	R	See [CFG.SPR2]
0x43619 - 0x4361A	sPR Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4361B - 0x4361C	sPR Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)
0x4361D - 0x4361E	sPRc Tuning Low	f32	R/W	default: 0.0 (Hz)
0x4361F - 0x43620	sPRc Tuning High	f32	R/W	default: 3333.3333 / 800.0 (Hz)

6.4.7 Data Recorder

Reg. Addresses	Description	Type	R/W	Notes
0x44000	Record Type	u16	R/W	See [CFG.DR1]
0x44001 - 0x44002	Time to Record	f32	R/W	See [CFG.DR2]
0x44003 - 0x44004	Samples to Record	u32	R/W	See [CFG.DR2]
0x44005 - 0x44006	Record Index	u32	R/W	See [CFG.DR3]
0x44007	Start Recording	u16	R/W	See [CFG.DR4]
0x44008	Stop Recording	u16	R/W	See [CFG.DR5]

[CFG.DR1] Data Recorder Record Type

Default value: 0x0001 AE sensor

Raw Data Record Types:

- 0x0001 AE sensor
- 0x0011 (combined) AE sensor, accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
- 0x0012 (combined) accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
- 0x0021 (combined) accelerometer all channel (XYZ, X, Y, Z)
- 0x0022 Accelerometer XYZ channel
- 0x0023 Accelerometer X channel
- 0x0024 Accelerometer Y channel
- 0x0025 Accelerometer Z channel
- 0x0031 (combined) gyroscope all channel (XYZ, X, Y, Z)
- 0x0032 Gyroscope XYZ channel
- 0x0033 Gyroscope X channel
- 0x0034 Gyroscope Y channel
- 0x0035 Gyroscope Z channel

PSD Array Record Types:

- 0x0201 AE sensor
- 0x0222 Accelerometer XYZ channel
- 0x0223 Accelerometer X channel
- 0x0224 Accelerometer Y channel
- 0x0225 Accelerometer Z channel
- 0x0232 Gyroscope XYZ channel
- 0x0233 Gyroscope X channel
- 0x0234 Gyroscope Y channel
- 0x0235 Gyroscope Z channel

[CFG.DR2] Data Recorder Time to Record, Samples to Record

Raw data record types:

- “Time to Record” and “Samples to Record” depend on each other. Only one of the parameters needs to be set, the other one is set automatically

- “Time to Record”: default: 87.3813 (s)
- “Samples to Record”: 8388608
- Parameter for single raw data sources
    - “Record Types”:
        - AE sensor
        - Accelerometer XYZ channel
        - Accelerometer X channel
        - Accelerometer Y channel
        - Accelerometer Z channel
        - Gyroscope XYZ channel
        - Gyroscope X channel
        - Gyroscope Y channel
        - Gyroscope Z channel
    - Max “Samples to Record”: 8388608
    - Max “Time to Record”: Max “Samples To Record” / sample rate (eg. AE sensor: 8388608 / 96000 = 87.38 (s))
- Parameter for combined raw data sources
    - “Record Types”:
        - (combined) AE sensor, accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
        - (combined) accelerometer all channel (XYZ, X, Y, Z), gyroscope all channel (XYZ, X, Y, Z)
        - (combined) accelerometer all channel (XYZ, X, Y, Z)
        - (combined) gyroscope all channel (XYZ, X, Y, Z)
    - It is recommended to set the parameter for the combined raw data record types via “Time to Record”
    - “Time to Record” is “Time to Record” for each data source of combined record type
    - “Samples to Record” is combined number of samples to record and depends on sample rates of data sources and “Time to Record”

PSD array record types:

- “Time to Record” and “Samples to Record” can’t be set for PSD array record types and are pre-set by data recorder module if PSD array record type gets selected
- Both depend on Sample Rate and Window Length for selected sensor data source of PSD array
- “Time to Record” is the Duration D of each PSD calculation
- “Samples to Record” is the number of output samples of the PSD. Example: If block length (number of input samples) of the PSD is 4096, the “Samples to Record” will be 2048

[CFG.DR3] Data Recorder Record Index
Record index of Data[0] at data recorder measurement registers can be reset via this register

[CFG.DR4] Data Recorder Start Recording
Write ‘1’ to start recording with selected “Record Type”, “Time to Record” and “Samples to Record”.
This register stays at ‘1’ as long as recording is ongoing and resets to ‘0’ if recording is finished. While recording is running, no new recording can be started and “Record Type”, “Time to Record” and “Samples to Record” can’t be changed.
If you need to change the configuration, write ‘1’ to “Data Recorder Stop Recording” to stop current recording

[CFG.DR5] Data Recorder Stop Recording
Write ‘1’ to stop recording.
If you stop a recording, already recorded data can be still be read

6.4.8 Autonomous Condition Monitoring

6.4.8.1 Condition Monitor 1

Reg. Addresses	Description	Type	R/W	Notes
0x45000	CM 1 Mode	u16	R/W	see [CFG.CM1]
0x45001	CM 1 Data Source Reg.	u16	R/W	see [CFG.CM2]
0x45002 - 0x45003	CM 1 Lower Threshold	16/32	R/W	see [CFG.CM3]
0x45004 - 0x45005	CM 1 Upper Threshold	16/32	R/W	see [CFG.CM3]
0x45006	CM 1 Alert Output	u16	R/W	see [CFG.CM4]
0x45007	CM 1 Output Type	u16	R/W	see [CFG.CM5]
0x45008	CM 1 GPIO Mode	u16	R/W	see [CFG.CM6]
0x45009 - 0x4500A	CM 1 1st Alert Index	u32	R	see [CFG.CM7]
0x4500B - 0x4500C	CM 1 1st Alert Value	16/32	R	see [CFG.CM8]

[CFG.CM1] Condition Monitoring - Mode
0 Off (default), 1 above value alert, 2 below value alert, 3 in range alert, 4 out of range alert

[CFG.CM2] Condition Monitoring - Data Source Register
Select any address from the measurement registers section.
If value is data type uint32/float32, selected MSB (= first) address

[CFG.CM3] Condition Monitoring - Lower/Upper Threshold
When a (u)int16 measurement register is used as data source register, a (u)int16 threshold must be configured. The (u)int16 threshold is the MSB (=first) part of the lower/upper threshold value.
When data type is uint32/float32 both register addresses are used.

[CFG.CM4] Condition Monitoring - Alert Output
0 Off (default), 1 USER LED (green/red), 2 GPIO Pin 1, 3 GPIO Pin 2

[CFG.CM5] Condition Monitoring - Output Type
0 Switched (default), 1 Latched
Output type if alert is active. Switched - automatically reset alert output, latched - manually reset alert output

[CFG.CM6] Condition Monitoring - GPIO Mode
0 high active (default), 1 low active

[CFG.CM7] Condition Monitoring - First Alert Index
When an alert is activated for the first time, the device will save the current “AE Measurement Index” (independent of the actual data source register). With knowledge of the current “AE Window Length”, the user can approximately calculate the time since the alert was activated for the first time.

[CFG.CM8] Condition Monitoring - First Alert Value
When an alert is activated for the first time, the device will save the current measurement register value, which triggered the alert, in this register.

The value will be same format as the monitored measurement register: int16, uint16, uint32, float32

6.4.8.2 Condition Monitor 2

Reg. Addresses	Description	Type	R/W	Notes
0x45010	CM 2 Mode	u16	R/W	see [CFG.CM1]
0x45011	CM 2 Data Source Reg.	u16	R/W	see [CFG.CM2]
0x45012 - 0x45013	CM 2 Lower Threshold	16/32	R/W	see [CFG.CM3]
0x45014 - 0x45015	CM 2 Upper Threshold	16/32	R/W	see [CFG.CM3]
0x45016	CM 2 Alert Output	u16	R/W	see [CFG.CM4]
0x45017	CM 2 Output Type	u16	R/W	see [CFG.CM5]
0x45018	CM 2 GPIO Mode	u16	R/W	see [CFG.CM6]
0x45019 - 0x4501A	CM 2 1st Alert Index	u32	R	see [CFG.CM7]
0x4501B - 0x4501C	CM 2 1st Alert Value	16/32	R	see [CFG.CM8]

6.4.8.3 Condition Monitor 3

Reg. Addresses	Description	Type	R/W	Notes
0x45020	CM 3 Mode	u16	R/W	see [CFG.CM1]
0x45021	CM 3 Data Source Reg.	u16	R/W	see [CFG.CM2]
0x45022 - 0x45023	CM 3 Lower Threshold	16/32	R/W	see [CFG.CM3]
0x45024 - 0x45025	CM 3 Upper Threshold	16/32	R/W	see [CFG.CM3]
0x45026	CM 3 Alert Output	u16	R/W	see [CFG.CM4]
0x45027	CM 3 Output Type	u16	R/W	see [CFG.CM5]
0x45028	CM 3 GPIO Mode	u16	R/W	see [CFG.CM6]
0x45029 - 0x4502A	CM 3 1st Alert Index	u32	R	see [CFG.CM7]
0x4502B - 0x4502C	CM 3 1st Alert Value	16/32	R	see [CFG.CM8]

6.4.8.4 Condition Monitor 4

Reg. Addresses	Description	Type	R/W	Notes
0x45030	CM 4 Mode	u16	R/W	see [CFG.CM1]
0x45031	CM 4 Data Source Reg.	u16	R/W	see [CFG.CM2]
0x45032 - 0x45033	CM 4 Lower Threshold	16/32	R/W	see [CFG.CM3]
0x45034 - 0x45035	CM 4 Upper Threshold	16/32	R/W	see [CFG.CM3]
0x45036	CM 4 Alert Output	u16	R/W	see [CFG.CM4]
0x45037	CM 4 Output Type	u16	R/W	see [CFG.CM5]
0x45038	CM 4 GPIO Mode	u16	R/W	see [CFG.CM6]
0x45039 - 0x4503A	CM 4 1st Alert Index	u32	R	see [CFG.CM7]
0x4503B - 0x4503C	CM 4 1st Alert Value	16/32	R	see [CFG.CM8]

6.4.8.5 Condition Monitor 5

Reg. Addresses	Description	Type	R/W	Notes
0x45040	CM 5 Mode	u16	R/W	see [CFG.CM1]
0x45041	CM 5 Data Source Reg.	u16	R/W	see [CFG.CM2]
0x45042 - 0x45043	CM 5 Lower Threshold	16/32	R/W	see [CFG.CM3]
0x45044 - 0x45045	CM 5 Upper Threshold	16/32	R/W	see [CFG.CM3]
0x45046	CM 5 Alert Output	u16	R/W	see [CFG.CM4]
0x45047	CM 5 Output Type	u16	R/W	see [CFG.CM5]
0x45048	CM 5 GPIO Mode	u16	R/W	see [CFG.CM6]
0x45049 - 0x4504A	CM 5 1st Alert Index	u32	R	see [CFG.CM7]
0x4504B - 0x4504C	CM 5 1st Alert Value	16/32	R	see [CFG.CM8]

6.4.8.6 Condition Monitor Logic A

Reg. Addresses	Description	Type	R/W	Notes
0x45100	CM Logic A Mode	u16	R/W	see [CFG.CM9]
0x45101	CM Logic A Source 1	u16	R/W	see [CFG.CM10]
0x45102	CM Logic A Source 2	u16	R/W	see [CFG.CM10]
0x45103	CM Logic A Alert Out.	u16	R/W	see [CFG.CM4]
0x45104	CM Logic A Out. Type	u16	R/W	see [CFG.CM5]
0x45105	CM Logic A GPIO Mode	u16	R/W	see [CFG.CM6]
0x45106 - 0x45107	CM Logic A 1st Al. I.	u32	R	see [CFG.CM7]

[CFG.CM9] Condition Monitoring Logic - Mode
0 Off (default), 1 AND, 2 OR, 3 XOR, 4 NOR, 5 XNOR, 6 NAND

[CFG.CM10] Condition Monitoring Logic - Data Source Register
Select any address from the status coil registers section.
Suggestion: when using combination between CM logic x’s, configure the output at CM logic C

6.4.8.7 Condition Monitor Logic B

Reg. Addresses	Description	Type	R/W	Notes
0x45110	CM Logic B Mode	u16	R/W	see [CFG.CM9]
0x45111	CM Logic B Source 1	u16	R/W	see [CFG.CM10]
0x45112	CM Logic B Source 2	u16	R/W	see [CFG.CM10]
0x45113	CM Logic B Alert Out.	u16	R/W	see [CFG.CM4]
0x45114	CM Logic B Out. Type	u16	R/W	see [CFG.CM5]
0x45115	CM Logic B GPIO Mode	u16	R/W	see [CFG.CM6]
0x45116 - 0x45107	CM Logic B 1st Al. I.	u32	R	see [CFG.CM7]

6.4.8.8 Condition Monitor Logic C

Reg. Addresses	Description	Type	R/W	Notes
0x45120	CM Logic C Mode	u16	R/W	see [CFG.CM9]
0x45121	CM Logic C Source 1	u16	R/W	see [CFG.CM10]
0x45122	CM Logic C Source 2	u16	R/W	see [CFG.CM10]
0x45123	CM Logic C Alert Out.	u16	R/W	see [CFG.CM4]
0x45124	CM Logic C Out. Type	u16	R/W	see [CFG.CM5]
0x45125	CM Logic C GPIO Mode	u16	R/W	see [CFG.CM6]
0x45126 - 0x45107	CM Logic C 1st Al. I.	u32	R	see [CFG.CM7]

Declaration of Conformity

- CE compliant
- RoHS compliant
- 3TG compliant
- REACH compliant

The smartPREDICT-AE is a component, which contains lead zirconium titanate in a proportion of more than 0.1% by weight.

Lead zirconium titanate (abbreviation: PZT, Lead titanium zirconium oxide) has been included in the SVHC list by the European Chemicals Agency (ECHA) (EC No: 235-727-4, CAS No: 12626-81-2).

iNDTact GmbH does not manufacture PZT itself but purchases it in crystalline and insoluble form from suppliers who have registered the substance with the ECHA. If the device is used as intended, no release of PZT can occur at any time. Risks can only occur in the case of inhalation or oral uptake (e.g. in the form of dust after mechanical processing or machining), but these can be ruled out if used as intended.

Legal Disclaimer

Pre-Series Versions (engineering samples)
Herein described is a pre-series version of the smartPREDICT-AE product. These versions are seen as engineering samples and may vary from the valid technical specifications of the product series contained in this data sheet. They are therefore not intended or fit for resale to third parties or for use in end products. Their sole purpose is internal client testing. The testing of an engineering sample may in no way replace the testing of a product series. iNDTact assumes no liability for the use of samples. The purchaser shall indemnify iNDTact from all claims arising from the use of pre-series versions.

Product Use
iNDTact products may only be used within the parameters of this product data sheet. They are not fit for use in life- sustaining or security sensitive systems. Security sensitive systems are those for which a malfunction is expected to lead to bodily harm or significant property damage. The resale and / or use of products are at the purchaser´s own risk and his own responsibility. The examination of fitness for the intended use is the sole responsibility of the purchaser. The purchaser shall indemnify iNDTact from all third party claims arising from any product use not covered by the parameters of this product data sheet or not approved by iNDTact and reimburse iNDTact for all costs in connection with such claims. The purchaser must monitor the market for the purchased products, particularly with regard to product safety, and inform iNDTact without delay of all security relevant incidents.

Application Examples
With respect to any examples given herein, any typical values stated herein and/ or any information regarding the application of the device, iNDTact hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights or copyrights of any third party. The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. They are provided for illustrative purposes only and no evaluation regarding infringement of intellectual property rights or copyrights or regarding functionality, performance or error has been made.

Contact Information

iNDTact GmbH
Friedrich-Bergius-Ring 15
97076 Würzburg
Germany

Tel.: +49 (0) 931 2999 7 330
Fax.: +49 (0) 931 2999 7 325
Email: info@indtact.de
Website: www.indtact.de