



MODBUS PROTOCOL

SPECIFICATIONS FOR SISGEO DIGITAL
INSTRUMENTS WITH FW VERSION 3 OR
HIGHER AND IN "NORMAL MODE"

USER MANUAL

INDEX

1. Introduction	3
2. Conversion of readings	7
3. MODBUS RS485 Communication	13
4. Legacy mode operation	30
5. Reading a SISGEO digital instrument	31

1. Introduction

The purpose of this manual is to describe the Modbus protocol used by SISGEO digital instruments. It is considered that the user already knows the MODBUS standard and its serial line implementation; all documents related to the standard can be freely downloaded from the www.modbus.org web site.

The documents of interest are:

- MODBUS Protocol Specification
- MODBUS Serial Line Protocol and Implementation Guide V1.02

In the last chapter “Reading a SISGEO digital instrument” it is described how to read a SISGEO digital instrument step by step.

Each SISGEO digital instrument has its own peculiarities, but normally it’s provided with an electronic board with the following main features:

1. High performance, low power microcontroller with memory
2. A Precision Analog-to-Digital Converter for single ended or differential analog input channels (sensors); also, digital input channels (SPI, etc.) can be read
3. Analog and digital power supply (sensors)
4. On board Temperature measurement
5. On board Humidity measurement
6. RS485 MODBUS communication with RTU protocol
7. Flexible power supply with on board supply voltage monitor
8. Low power operation

It should be assembled a termination resistance on the first and the last RS485 node on the linear RS485 bus; a node can either be an instrument or the master controlling the bus.

Note: the firmware of the electronic board can be upgraded by SISGEO: please contact Sisgeo’s Customer Care Department (<https://support.sisgeo.com/>) in order to request an RMA ticket for upgrading the MODBUS firmware revision.

At the time the present manual is issued, SISGEO has two different hardware version of digital electronic boards that includes different firmware versions.

In each firmware is also configured the operating mode ‘Legacy’ or ‘Normal’.

SISGEO utilizes ‘Legacy’ mode in order to make the new digital instrument compatible for the Modbus master not yet upgraded to firmware version 3.

This manual deals with Modbus protocol for SISGEO digital instruments implemented with firmware ver. 3.x in Normal mode.

For SISGEO digital instruments implemented with firmware ver. 0.0, ver. 1 (Legacy mode), ver. 2 (Legacy mode) and ver. 3 (Legacy mode) the reference document is “SISGEO_digital_instruments_-_Modbus_protocol_specification_EN_04_17” available on SISGEO web site.



The first version of SISGEO digital instruments, utilized until July 2017, employs firmware from version 0.0 up to version 2.3. This firmware uses Legacy mode.

The second version of SISGEO digital instruments, utilized starting from June 2017, and still used, employs firmware from version 3.5 onward; this firmware can use both Legacy mode and Normal mode.

SISGEO started to supply some instruments in Normal mode from July 2019.

Hardware and firmware of each SISGEO digital instrument can be found in the Calibration Report, together with the RS485 Modbus address and the use of Timed mode for the time-controlled switch on, where applicable.



CALIBRATION REPORT

N. 2001307

Serial Number: D200335

Product Code: 0DEX35S115D_S

Model: DIGITAL DEX-S ESTENSO-INCLIN. FIJO

Type: Digital

Calibration Date: 10/02/2020 14:37

Job Number: 20-00099

Project Code: Address #01 Timed 3-3

FW/HW Ver. 3.17/ -

The configured mode (Legacy or Normal) instead is not usually written in the Calibration Report. Please contact the SISGEO Customer Care Department in case you are not able to obtain this information.

Below a list of our SISGEO digital instruments:

Description	Number of measurement channels	Unit of measurement (configurable in the factory)	Troubleshooting: temperature, humidity, power supply voltage
BH-profile, In-Place-Inclinometer	1 (uniaxial) or 2 (biaxial)	° (degree), $\sin\alpha$, $20k\sin\alpha$, mm/m, inches/feet, $(k)^* \sin\alpha$ (where k could be between 2 and 100k)	available
Tiltmeter	1 (uniaxial) or 2 (biaxial)	° (degree), $\sin\alpha$, $20k\sin\alpha$, mm/m, inches/feet, $(k)^* \sin\alpha$ (where k could be between 2 and 100k)	available
Railway Deformation System (RDS)	1	° (degree), $\sin\alpha$, $20k\sin\alpha$, mm/m, inches/feet, $(k)^* \sin\alpha$ (where k could be between 2 and 100k)	available

Settlement Gauge (H-LEVEL)	1	mmH ₂ O	available
Extensometer (DEX)	1	mm, feet, inch	available
Extenso-inclinometers (DEX-S)	3	2 first channels (inclinometers) configured in: ° (degree), $\sin\alpha$, $20k\sin\alpha$, mm/m, inches/feet, $(k)^* \sin\alpha$ (where k could be between 2 and 100k). The last one channel (extensometer) configured in in 'mm'.	available
Load cell	1	KN, tonne, long ton, short ton	available
Piezometers	1	Bar, mbar, atm, psi, Pa, kPa, MPa, mmH ₂ O, mH ₂ O, inH ₂ O, ftH ₂ O, mmHg, cmHg, inHg, Kg/cm, Kg/m, lb/in, lb/ft ² , N/cm, N/m ² , t/m ² , t(UK)/ft ² , t(USA)/ft ²	available
DIGITAL SINGLE POINT ROD EXTENSOMETER	1	mm, feet, inch	available
DIGITAL 2-BASE POINT ROD EXTENSOMETER	2	mm, feet, inch	available
DIGITAL 3-BASE POINT ROD EXTENSOMETER	3	mm, feet, inch	available
DIGITAL 4-BASE POINT ROD EXTENSOMETER	4	mm, feet, inch	available
DIGITAL 5-BASE POINT ROD EXTENSOMETER	5	mm, feet, inch	available
DIGITAL 6-BASE POINT ROD EXTENSOMETER	6	mm, feet, inch	available
DIGITAL 7-BASE POINT ROD EXTENSOMETER	7	mm, feet, inch	available



For the T-1000 telecoordinometer please refer to its User Manual.

Each SISGEO digital instruments feature can be found in its own Technical Datasheet and User Manual: we recommend to check these documents in addition to the present Manual.

Every SISGEO digital instrument can work in two modes: ALWAYS ON and TIMED.
SISGEO Communications Window Feature (Timed mode) works both in Normal mode and in Legacy mode.

Standard configuration of SISGEO digital instruments is ALWAYS ON (except DEX and DEX-S).
Timed mode is used on request.

In order to be sure that a new measurement has been performed, we suggest to wait for the COUNT register to change: always use the COUNT register for this, not the CHANGE register that has a different meaning.

Presently, for static measurement, in order to correctly read a SISGEO digital instruments, we recommend to wait until 3, or more, full reading have been completed (count register), so that the measure will be stable.

2. Conversion of readings

During normal operations, the electronic board reads raw data from the AD converter or directly from another digital sensor and converts it into the physical measure, (for example a tilt or a pressure).

Depending on the type of measure being obtained, the process can be more or less complicated, but due to the fact that the physical measure is taken through an electrical device (e.g. an AD converter), the process always takes two steps:

1. The conversion of the raw bit value (counts) read into an electrical value (e.g. a value in mV)
2. The conversion of the electrical value into the physical measure (e.g. a value in sine α or in kPa)

Splitting the conversion in two steps brings the advantage that the calibration can also be divided into two different independent parts:

1. The electronic board calibration in which the connection between counts and the electrical value is defined
2. A transducer calibration in which the characteristics of the sensor are taken into account.

This allows to replace electronic board and sensor independently.

Thus, when performing a measure, we have the following flux of data:

Raw bit value → Electrical value → Physical measure

It must be observed that, while in the first step the single acquisition channels are treated separately, in the second step the channels can interact with one another, as, for example, with a biaxial inclinometer in which the milliVolts read for each axis are used to calculate the real physical values and corrected for cross axes and other interactions.

In this process, the number of physical values can also end up to be different from the number of electrical values.

Accordingly, we have the following groups of MODBUS Input Registers:

IR_RAW_...	These registers contain the RAW reading from the AD converters
IR_ELECTRICAL_...	These registers contain the electrical value for the channel of the AD converter in floating point format
IR_PHYSICAL_...	These registers contain the physical value in floating point
IR_PHYSICALINT_...	These registers contain the physical value in integer or fixed-point format, typically to allow Legacy application to access a 16-bit value in the most significant part of each register.

Some internal sensors (e.g. Humidity and Temperature sensor), although they obviously use some kind of physical to electrical conversion internally, do not expose this value to the outside world, and they only offer a default conversion parameter from raw to physical readings. In this case the default conversion is used and the approximate physical value is shown as the electrical value; it is possible improve the conversion when calibrating the sensor in the Electrical to Physical conversion step.

Different types of Converters

The electronic board supports different type of conversion both on board and external; different conversions are handled for each of them.

The electronic board supports different raw and electrical channels and each channel can be configured independently to source data from any converter: they can, and normally are, be used as source for more than one channel, as most converters contain a multiplexer.

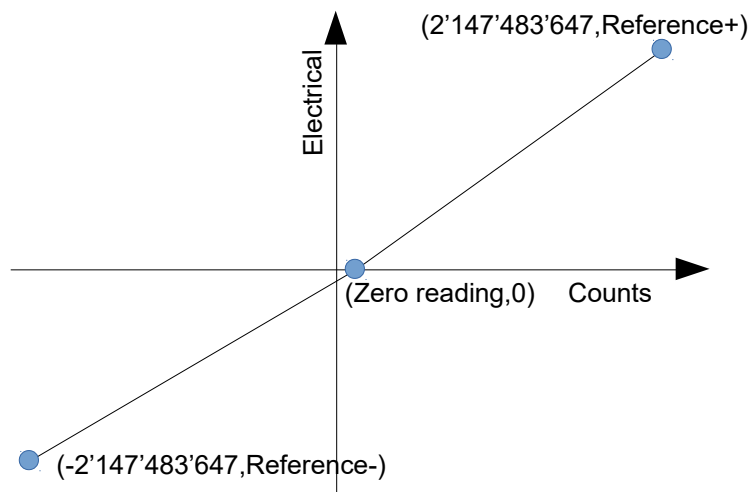
The configuration of each electrical channel depends on the sensor that is used as data source.

Precision Analog to Digital Converters

The AD converter can be configured to operate at different speed.

The linearization from counts to the electrical unit is handled to convert from:

- IR_RAW, for example a value in the $\pm 2'147'483'647$ ($\pm 0x7FFF'FFFF$) range, to
- IR_ELECTRICAL: a value that can range from Reference- to Reference+ although the actual input range is normally kept short of the maximum theoretical range.



The conversion is linear in both ranges (Zero reading, 2'147'483'647) and (- 2'147'483'647, Zero reading). The two lines can have slightly different slopes.

Humidity and Temperature Sensor

This converter is used to measure temperature and humidity inside the instrument.

“Temperature” can be used to compensate temperature sensitive sensors.

“Humidity” is used to detect water leaking inside the instrument.

The conversion to electrical unit is fixed, so it is only possible to specify if the given raw/electrical channel should source temperature or humidity.

It is possible to adjust the conversion by inserting calibration parameters in the conversion from IR_ELECTRICAL to IR_PHYSICAL.

Supply voltage measurement

This converter measures the supply voltage; the IR_RAW value is in a specific range (for example 0 to 4095); then the value is converted to IR_ELECTRICAL with a specific formula.

The averaging process

Different averaging processes can be used in order to filter noise as fixed or moving.

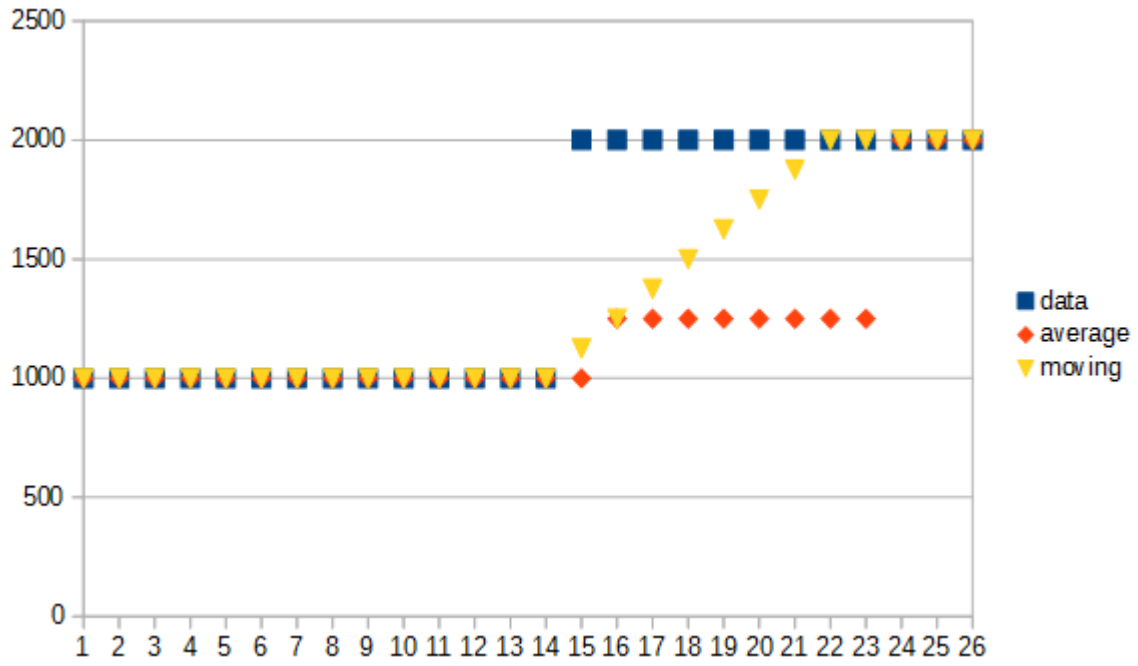
The averaging process can be configured independently for each raw IR_RAW channel.

[0x0148]	Warmup time	<input type="text" value="10"/>	Tenth of seconds
[0x0149]	Average	<input type="text" value="50"/>	
[0x014A]	Moving average	<input type="text" value="4"/>	
[0x014B]	Cooling down	<input type="text" value="0"/>	Tenth of seconds

The average is performed during the acquisition of the IR_RAW data: the requested number of readings of the AD converter is performed and averaged, so that the IR_RAW value read is actually the average of the readings.

When more channels are defined, the requested number of reading is accumulated before switching to the new channel.

The moving average, instead, is performed on the IR_RAW data; each new value read replaces the oldest data read for the same channel and the average is calculated.



Input	Average	Queue for moving average				Moving Average
		Oldest		Last		
0						
1						
2						
3						
4	2	2	2	2	2	2
5						
6						
7						
8						
9	7	2	2	2	7	3.25
10						
11						
12						
13						
14	12	2	2	7	12	5.75
15						
16						
17						
18						
19	17	2	7	12	17	9.5
19						
19						
19						
19	19	7	12	17	19	13.75
19						
19						
19						
19	19	12	17	19	19	16.75
19						
19						
19						
19	19	17	19	19	19	18.5
19						
19						
19						
19	19	19	19	19	19	19

Note: the ADC normally has a programmable acquisition rate: decreasing the acquisition rate increases the Effective Number Of Bits (ENOB) and reduces the sensitivity to noise. Reducing the acquisition rate can be more effective than increasing the number of readings for fixed and moving average.

Other channel parameters

[0x014C-0x014D]	Multiplier to fixed point	<input type="text" value="1"/>
[0x014E]	Model	<input type="text" value="Generic"/>

The value read from the IR_PHYSICAL_INT_CHANNEL_0 to IR_PHYSICAL_INT_CHANNEL_7 is by default the same value from IR_PHYSICAL_CHANNEL_0 to IR_PHYSICAL_CHANNEL_7 rounded to an integer; to get more significant digits it is possible to define a multiplying factor for each channel so that:

$IR_PHYSICAL_INT_CHANNEL = ROUND (IR_PHYSICAL_CHANNEL * multiplier)$

For example, if IR_PHYSICAL_CHANNEL_0 reads 3.1415927, with the default value of 1 IR_PHYSICAL_INT_CHANNEL_0 would read 3; setting multiplier for channel 0 to 1000 IR_PHYSICAL_INT_CHANNEL_0 would read 3141.

For each channel you can define a register containing the model of the instrument connected to the channel; this can be used as a reference by the logger reading the unit.

The conversion from electrical to physical measure

The conversion from electrical to physical measure is handled by algorithms embedded in the electronic board; this algorithm reads its input from the available electrical channels and stores results in physical channels.

3. MODBUS RS485 Communication

Each SISGEO digital instrument implements the MODBUS protocol on an RS485 interface. The default baud rate is 9600, but it is possible to configure the unit to operate at different baud rates setting the desired baud rate in the User configuration. The communication format is 8 data bits, 1 stop bit, parity none.

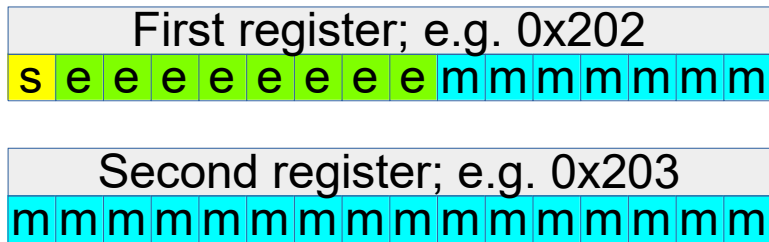
Normal use and configuration are handled through MODBUS input registers and MODBUS Holding registers.

Please, pay attention to the difference between Input and Holding registers.

The following registers contain data in IEEE 754 binary 32 floating point format:

- IR_PHYSICAL_CHANNEL_0 to IR_PHYSICAL_CHANNEL_14
- IR_ELECTRICAL_CHANNEL_0 to IR_ELECTRICAL_CHANNEL_14

The order of the data in the registers is as follows:



NOTE For further details on IEEE 754 binary 32 format check:

https://en.wikipedia.org/wiki/IEEE_floating_point.

Reading MODBUS registers in sequence is faster than reading registers at random addresses; for this reason some registers have been duplicated:

- IR_PHYSICAL_COUNT, IR_PHYSICALINT_COUNT, IR_ELECTRICAL_COUNT and IR_RAW_COUNT all read the same value as well as IR_LEGACY_COUNT.
- IR_PHYSICAL_CHANGE, IR_PHYSICALINT_CHANGE, IR_ELECTRICAL_CHANGE and IR_RAW_CHANGE all read the same value.

This allows the updated status and the relevant data to be read in a single transaction.

Input registers for physical values

The following registers hold the physical values read by SISGEO digital instrument.

Address	Name
0x200	IR_PHYSICAL_COUNT
0x201	IR_PHYSICAL_CHANGE
0x202	IR_PHYSICAL_CHANNEL_0
0x204	IR_PHYSICAL_CHANNEL_1
0x206	IR_PHYSICAL_CHANNEL_2
0x208	IR_PHYSICAL_CHANNEL_3
0x20A	IR_PHYSICAL_CHANNEL_4
0x20C	IR_PHYSICAL_CHANNEL_5
0x20E	IR_PHYSICAL_CHANNEL_6
0x210	IR_PHYSICAL_CHANNEL_7
0x212	IR_PHYSICAL_CHANNEL_8
0x214	IR_PHYSICAL_CHANNEL_9
0x216	IR_PHYSICAL_CHANNEL_10
0x218	IR_PHYSICAL_CHANNEL_11
0x21A	IR_PHYSICAL_CHANNEL_12
0x21C	IR_PHYSICAL_CHANNEL_13
0x21E	IR_PHYSICAL_CHANNEL_14

IR_PHYSICAL_COUNT

This register is incremented each time the readings in all IR_PHYSICAL_CHANNEL registers have been updated.

IR_PHYSICAL_CHANGE

This register is incremented each time all the moving averages have been completely updated with new values.

IR_PHYSICAL_CHANNEL_0 to IR_PHYSICAL_CHANNEL_14

These registers hold the physical reading for all the channels; each reading in floating point format, uses two registers; e.g. IR_PHYSICAL_CHANNEL_0 uses two registers (0x202 and 0x203).

The values are in IEEE 754 binary 32 floating point format.

Input registers for physical values (fixed point)

The following registers hold the physical values read by the SISGEO digital instrument, but in fixed point format.

Address	Name
0x220	IR_PHYSICAL_INT_COUNT
0x221	IR_PHYSICAL_INT_CHANGE
0x222	IR_PHYSICAL_INT_CHANNEL_0
0x224	IR_PHYSICAL_INT_CHANNEL_1
0x226	IR_PHYSICAL_INT_CHANNEL_2
0x228	IR_PHYSICAL_INT_CHANNEL_3
0x22A	IR_PHYSICAL_INT_CHANNEL_4
0x22C	IR_PHYSICAL_INT_CHANNEL_5
0x22E	IR_PHYSICAL_INT_CHANNEL_6
0x230	IR_PHYSICAL_INT_CHANNEL_7
0x232	IR_PHYSICAL_INT_CHANNEL_8
0x234	IR_PHYSICAL_INT_CHANNEL_9
0x236	IR_PHYSICAL_INT_CHANNEL_10
0x238	IR_PHYSICAL_INT_CHANNEL_11
0x23A	IR_PHYSICAL_INT_CHANNEL_12
0x23C	IR_PHYSICAL_INT_CHANNEL_13
0x23E	IR_PHYSICAL_INT_CHANNEL_14

IR_PHYSICAL_INT_COUNT

This register is incremented each time the readings in all the IR_PHYSICAL_CHANNEL registers have been updated.

IR_PHYSICAL_INT_CHANGE

This register is incremented each time all the moving averages have been completely updated with new values.

IR_PHYSICAL_INT_CHANNEL_0 to IR_PHYSICAL_INT_CHANNEL_14

These registers contain the same data as IR_PHYSICAL_CHANNEL_0 to IR_PHYSICAL_CHANNEL_14, but in integer or fixed point format to simplify access to data from systems that do not implement floating point operations.

The value in the corresponding IR_PHYSICAL_CHANNEL register is multiplied by a constant contained in the configuration and the result is converted into a signed 32 bit integer number; the result, in integer 2-complement format is stored in two successive registers (e.g. registers 0x202

and 0x203 for physical channel 0); the most significant word is stored in the first of the two registers, followed by the least significant part.

Input registers for electrical values

The following registers contain the electrical values read by the SISGEO digital instrument .

Address	Name
0x240	IR_ELECTRICAL_COUNT
0x241	IR_ELECTRICAL_CHANGE
0x242	IR_ELECTRICAL_CHANNEL_0
0x244	IR_ELECTRICAL_CHANNEL_1
0x246	IR_ELECTRICAL_CHANNEL_2
0x248	IR_ELECTRICAL_CHANNEL_3
0x24A	IR_ELECTRICAL_CHANNEL_4
0x24C	IR_ELECTRICAL_CHANNEL_5
0x24E	IR_ELECTRICAL_CHANNEL_6
0x250	IR_ELECTRICAL_CHANNEL_7
0x252	IR_ELECTRICAL_CHANNEL_8
0x254	IR_ELECTRICAL_CHANNEL_9
0x256	IR_ELECTRICAL_CHANNEL_10
0x258	IR_ELECTRICAL_CHANNEL_11
0x25A	IR_ELECTRICAL_CHANNEL_12
0x25C	IR_ELECTRICAL_CHANNEL_13
0x25E	IR_ELECTRICAL_CHANNEL_14

IR_ELECTRICAL_COUNT

This register is incremented each time the readings in all the IR_ELECTRICAL_CHANNEL registers have been updated.

IR_ELECTRICAL_CHANGE

This register is incremented each time all the moving averages have been completely updated with new values.

IR_ELECTRICAL_CHANNEL_0 to IR_ELECTRICAL_CHANNEL_14

These registers hold the electrical readings for all the channels; each reading, in floating point format, uses two registers; e.g. IR_ELECTRICAL_CHANNEL_0 uses two registers (0x242 and 0x243). The values are in IEEE 754 binary 32 floating point format.

Input registers for raw values

These registers contain the raw values read from the ADC converters in integer format; they are mainly intended for debugging the system and for obtaining the raw values needed for calculating the linearization parameters for the ADC converters.

Address	Name
0x260	IR_RAW_COUNT
0x261	IR_RAW_CHANGE
0x262	IR_RAW_CHANNEL_0
0x264	IR_RAW_CHANNEL_1
0x266	IR_RAW_CHANNEL_2
0x268	IR_RAW_CHANNEL_3
0x26A	IR_RAW_CHANNEL_4
0x26C	IR_RAW_CHANNEL_5
0x26E	IR_RAW_CHANNEL_6
0x260	IR_RAW_CHANNEL_7
0x262	IR_RAW_CHANNEL_8
0x264	IR_RAW_CHANNEL_9
0x266	IR_RAW_CHANNEL_10
0x268	IR_RAW_CHANNEL_11
0x26A	IR_RAW_CHANNEL_12
0x26C	IR_RAW_CHANNEL_13
0x26E	IR_RAW_CHANNEL_14

IR_RAW_COUNT

This register is incremented each time the readings in all the IR_RAW_CHANNEL registers have been updated.

IR_RAW_CHANGE

This register is incremented each time all the moving averages have been completely updated with new values.

IR_RAW_CHANNEL_0 to IR_RAW_CHANNEL_14

These registers contain the raw data from the ADC converters in integer 2-complement format; each value is stored in two successive registers (e.g. registers 0x262 and 0x263 for raw channel 0); the most significant word is stored in the first of the two registers, followed by the least significant part.

Legacy input registers

These registers are intended to ensure compatibility with SISGEO digital instrument ver. 0.x, ver. 1.x (Legacy mode), ver. 2.x (Legacy mode) and ver. 3.x (ONLY configured in Legacy mode).

Address	Name	Same as
0x0100	IR_LEGACY_COUNT	IR_PHYSICAL_COUNT
0x0101	IR_LEGACY_TYPE	N.A.
0x0110	IR_LEGACY_RAW_X	IR_RAW_CHANNEL_0
0x0112	IR_LEGACY_RAW_Y	IR_RAW_CHANNEL_1
0x0114	IR_LEGACY_RAW_T	IR_RAW_CHANNEL_2
0x0120	IR_LEGACY_FIX_X	IR_PHYSICAL_INT_0
0x0122	IR_LEGACY_FIX_Y	IR_PHYSICAL_INT_1
0x0124	IR_LEGACY_FIX_T	IR_PHYSICAL_INT_2
0x0126	IR_LEGACY_FLOAT_X	IR_PHYSICAL_0
0x0128	IR_LEGACY_FLOAT_Y	IR_PHYSICAL_1
0x012A	IR_LEGACY_FLOAT_T	IR_PHYSICAL_2

These registers are available both in Legacy and non-Legacy mode (Normal mode); when in Legacy mode the contents of these registers depend on the Legacy configuration; in any case there is a direct correspondence between these registers, except IR_LEGACY_TYPE, and the standard registers.

IR_LEGACY_TYPE

This register contains a bitmap that describes the characteristics of the previous SISGEO digital instruments being simulated by the new SISGEO digital instruments.

Refer to “SISGEO_digital_instruments_-_Modbus_protocol_specification_EN_04_17” for details.

Other input registers

These registers contain diagnostic information.

Address	Name
0x0400	IR_DIAGNOSTIC (RESERVED)
0x0401	IR_SUPPLY
0x0402	IR_UID_0
0x0403	IR_UID_1
0x0404	IR_UID_2
0x0405	IR_UID_3
0x0406	IR_UID_4
0x0407	IR_UID_5
0x0408	IR_TEMPERATURE
0x0409	IR_HUMIDITY

IR_DIAGNOSTIC

This register contains a bitmap for test results performed when the unit was last reset. This is a RESERVED register for SISGEO staff, only for debug operations.

IR_SUPPLY

This register contains the voltage supply in tenth of Volt; e.g. 242 means 24.2 Volts.

IR_UID_0 to IR_UID_5

These registers contain sixteen bit words that are unique to the SISGEO digital instrument: no SISGEO digital instrument share the same values. These values are factory programmed into the microprocessor of the unit and cannot be changed.

IR_TEMPERATURE

This register contains the temperature on the electronic board in tenths of Celsius: e.g. 215 means 21.5 °C.

IR_HUMIDITY

This register contains the relative humidity on the electronic board in tenths of percent: e.g. 403 means 40.3 %RH.

Version holding registers

These read only registers contain version information about different parts of the system.

Address	Name
0x0000	HR_VERSION_FW
0x0001	HR_VERSION_HW
0x0002	HR_VERSION_BIOS

HR_VERSION_FW

This register can be read to obtain the current firmware version mounted on the electronic board

HR_VERSION_HW

This register can be read to obtain the version of the hardware of the electronic board

HR_VERSION_BIOS

A BIOS is available that is used to bootstrap the system and to manage firmware upgrades; this register can be read to obtain the version of the BIOS.

User and Legacy configuration holding registers

These registers are used to hold the User configuration and the Legacy configuration.
The kind of configuration stored here depends on the Factory configuration.

Address	Name
0x0100	HR_USR_CFG_SIZE
0x0101	HR_USR_CFG_SIGNATURE
0x0102	HR_USR_OPTIONS
0x0103	HR_USR_CFG_ADDRESS
0x0104	HR_USR_CFG_SERIAL_1
0x0105	HR_USR_CFG_SERIAL_2
0x0106 to 0x0141	Reserved for Legacy configuration;-see "SISGEO_digital_instruments_-_Modbus_protocol_specification_EN_04_17" for details.
0x0142	HR_USR_CFG_BAUD
0x0148 to 0x014F	Channel 0 configuration
0x0150 to 0x0157	Channel 1 configuration
0x0158 to 0x015F	Channel 2 configuration
0x0160 to 0x0167	Channel 3 configuration
0x0168 to 0x016F	Channel 4 configuration
0x0170 to 0x0177	Channel 5 configuration
0x0178 to 0x017F	Channel 6 configuration
0x0180 to 0x0187	Channel 7 configuration
0x0188 to 0x018F	Channel 8 configuration
0x0190 to 0x0197	Channel 9 configuration
0x0198 to 0x019F	Channel 10 configuration
0x01A0 to 0x01A7	Channel 11 configuration

0x01A8 to 0x01AF	Channel 12 configuration
0x01B0 to 0x01B7	Channel 13 configuration
0x01B8 to 0x01BF	Channel 14 configuration

HR_USR_CFG_SIZE

This register contains the size of the configuration in bytes; the actual number of registers following this one in the configuration is (value of HR_USR_CFG_SIZE+1) / 2.

HR_USR_CFG_SIGNATURE

This register must contain the 0x5AA5 for valid configuration.

HR_USR_OPTIONS (TIMED)

The Options register (address 0x0102) is used to configure the SISGEO Communication Window Feature (Timed).

When this register is set to 0 (the default value) this feature is disabled and the SISGEO digital instruments is always on and ready to serve MODBUS commands.

When a different value is set, the most significant byte is the incremental delay parameter and the least significant byte is the initial delay parameter.

With a value of a register value of 0x050A, we would have:

incremental delay = 0x05 = 5
initial delay = 0x0A = 10

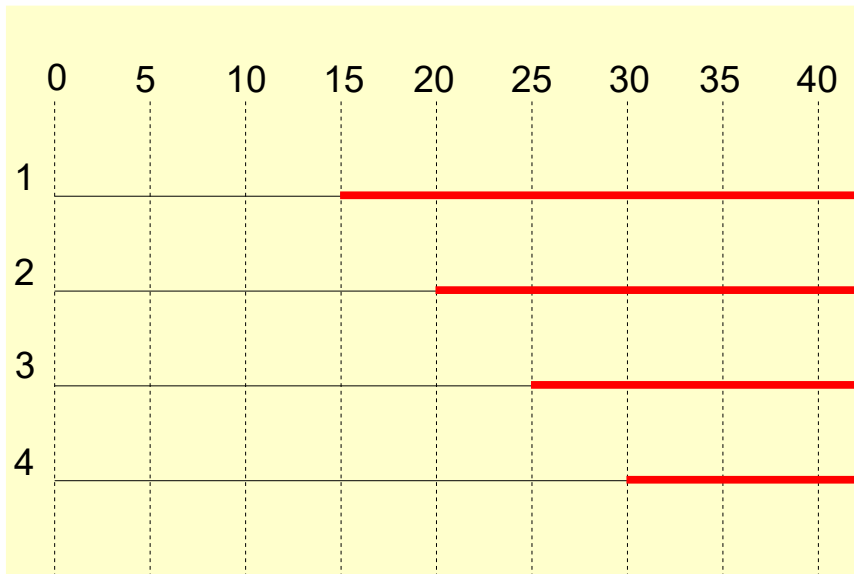
The SISGEO digital instruments will switch on after a time in seconds of:

$$delay = initialdelay + incrementaldelay * MODBUSaddress$$

So, units with addresses 1, 2, 3 and 4 would enable their communication interface after a delay of 15, 20, 25 and 30 when using the sample values above.

The MODBUS master should wait till the unit come online, read the required values and switch them off when they are no more needed.

The only way to switch on again a unit, after it had been switched off, is to cycle power off and on again.



HR_USR_CFG_ADDRESS

This register contains the MODBUS address used by the SISGEO digital instrument; although the value is a sixteen bit value, a MODBUS address must be restricted in the range 1 to 247.

HR_USR_CFG_SERIAL_1 and HR_USR_CFG_SERIAL_2

These registers can be read to obtain the serial number of the device. Writing to these registers has actually no effect.

These registers are actually a shadow copy of HR_MFG_SERIAL1 and HR_MFG_SERIAL2; refer to the description of these register for the format of the serial number.

HR_USR_CFG_BAUD

This register holds the baud rate used in communication divided by 100; e.g a value of 96 actually means a baud rate of 9600.

Channel configurations

Each channel configuration block contains the following registers; the address of each register must be obtained adding the register offset to the base address for the channel.

Offset	Name
+0	Warmup
+1	Average
+2	MovingAverage
+3	Cooldown
+4	Scale LSW
+5	Scale MSW
+6	Model

Warmup

Warmup time in tenth of second after the channel is configured, before measure actually starts.

Average

Number of readings to average for the channel before the value is passed to the moving average.

Moving Average

Number of entries in the moving average (a value of 0 or 1 here effectively disables the moving average, so a value of 1 is actually treated as 0). The total number of entries for all channels must be not above 64.

Cooldown

Time to wait after a reading in tenth of second before the same channel is read again.

Scale LSW and Scale MSW

Multiplier used on the floating-point value to obtain the integer representation:

$$IR_PHYSICAL_INT = IR_PHYSICAL \cdot Scale$$

Note: note that the order in which these registers are stored is different from the convention normally used in MODBUS of storing the most significant part first.

Model

A code describing the type of instrument connected to the channel; the following constants are defined.

Code	Model
0	Generic
1	IPI
2	Tiltmeter
3	Settlement Gauge
4	Pressure Cell
5	Load Cell
6	Piezometer

7	RDS
8	Pendulum
9	Extensometer
10	Crackmeter
11	Strain Gauge
12	Thermometer
13	Extensoinclinometer
14	T-1000 Telependulum

Writing configuration registers

In order to maintain compatibility with the SISGEO digital instrument ver. 0.x, ver. 1.x (Legacy mode), ver. 2.x (Legacy mode) and ver. 3.x (ONLY configured in Legacy mode) when writing configuration, the following sequence must be respected:

1. Read HR_USR_CFG_SIZE to determine the number of registers in configuration
2. Read and all the registers from 0x0101 up to $0x0101 + \langle \text{value of HR_USR_CFG_SIZE} + 1 \rangle / 2$
3. Modify values as needed
4. Write in HR_USR_CFG_SIZE the same value that was read
5. Write all the registers from 0x0101 up to $0x0101 + \langle \text{value of HR_USR_CFG_SIZE} + 1 \rangle / 2$ in sequence; registers can be written in different MODBUS transactions, but the sequence must be respected.

Note: a program for managing SISGEO digital instrument ver. 0 or ver. 1 (Legacy mode) configurations that *****wrongly***** assumed a fixed number of parameters would no longer work, as the number of parameters has been extended in ver. 3.

Factory configuration holding registers

These registers are used to hold the Factory configuration and are read only.

Address	Name
0x0200	HR_MFG_MODE
0x0201	HR_MFG_CONFIG_VERSION
0x0202	HR_MFG_SERIAL1
0x0202	HR_MFG_SERIAL2

HR_MFG_MODE

Describes the operating mode of the SISGEO digital instrument:

Code	Mode
0x5A01	Legacy configuration is used (ver. 0.x, ver. 1.x, ver. 2.x and ver. 3.x, the last one ONLY configured in Legacy mode)
0x5A02	Normal configuration is used (ver. 3.x)

HR_MFG_CONFIG_VERSION

This register contains the version of the configuration; this is different from the firmware version and is used to define the meaning of the configuration registers. The current value for this register is 0x0001.

When interpreting the configuration values, this value should be checked; if the value is different from 0x0001 you should check for an updated version of this document.

HR_MFG_SERIAL1 and HR_MFG_SERIAL2

These registers contain the serial number of the SISGEO digital instrument-

<i>HR_MFG_SERIAL1</i>			<i>HR_MFG_SERIAL2</i>			
<i>LSB</i>	<i>MSB</i>		<i>LSB</i>	<i>MSB</i>		
	<i>High nibble</i>	<i>Low nibble</i>	<i>High nibble</i>	<i>Low nibble</i>	<i>High nibble</i>	<i>Low nibble</i>
ASCII leading character of serial number	BCD[5]	BCD[4]	BCD[3]	BCD[2]	BCD[1]	BCD[0]

The numerical part of the serial number has the value:

$$100000 * \text{BCD}[5] + 10000 * \text{BCD}[4] + 1000 * \text{BCD}[3] + 100 * \text{BCD}[2] + 10 * \text{BCD}[1] + \text{BCD}[0]$$

For example, if HR_MFG_SERIAL1 had value 0x1253 and HR_MFG_SERIAL2 had value 0x5634, the value displayed should be ('S' has an ASCII value 0x53): 'S123456'.

Address	Name
0x0380	HR_MFG_PHYSICAL_UNIT_0
0x0381	HR_MFG_PHYSICAL_UNIT_0_AUX_0
0x0382	HR_MFG_PHYSICAL_UNIT_0_AUX_1
0x0383	HR_MFG_PHYSICAL_UNIT_0_AUX_2
0x0384 to 0x0387	Physical unit description for channel 1
0x0388 to 0x038B	Physical unit description for channel 2
0x038C to 0x038F	Physical unit description for channel 3
0x0390 to 0x0393	Physical unit description for channel 4
0x0394 to 0x0397	Physical unit description for channel 5
0x0398 to 0x039B	Physical unit description for channel 6
0x039C to 0x039F	Physical unit description for channel 7
0x03A0 to 0x03A3	Physical unit description for channel 8
0x03A4 to 0x03A7	Physical unit description for channel 9
0x03A8 to 0x03AB	Physical unit description for channel 10
0x03AC to 0x03AF	Physical unit description for channel 11
0x03B0 to 0x03B3	Physical unit description for channel 12
0x03B4 to 0x03B7	Physical unit description for channel 13
0x03B8 to 0x03BB	Physical unit description for channel 14

Description is shown for the first channel; the following channels replicate the same structure.

HR_MFG_PHYSICAL_UNIT_0 to HR_MFG_PHYSICAL_UNIT_14

These registers contain a constant that defines the physical unit measured by the channel; the constants are defined in the table below.

Code	Unit	Suggested number of decimals
0	Channel is not used	-
1	mV	2
2	bar	5
3	mbar	3
4	atm	5
5	psi	4
6	Pa	0
7	kPa	3
8	MPa	6
9	mmH2O	2
10	mH2O	5
11	inH2O	3
12	ftH2O	4
13	mmHg	3
14	cmHg	4
15	inHg	4
16	Kg/cm ²	5
17	Kg/m ²	1
18	lb/in ²	4
19	lb/ft ²	2
20	N/cm ²	4
21	N/m ²	0
22	t/m ²	4
23	t(UK)/ft ²	5
24	t(USA)/ft ²	5
25	°C	1
26	20kSina	2
27	Sina	6
28	° (Angle in Degrees)	4
29	%RH	1
30	mV/V	4
31	mm	4
32	mm/m	3
33	inches/feet	5
34	kSina	Max(0,6-floor(log10(k)))
35	Volt	1
36	kN	2
37	με	2
38	inch	5
39	feet	6
65535	Unit is undefined	-

HR_MFG_PHYSICAL_UNIT_0_AUX_0 to HR_MFG_PHYSICAL_UNIT_14_AUX_0
HR_MFG_PHYSICAL_UNIT_0_AUX_1 to HR_MFG_PHYSICAL_UNIT_14_AUX_1
HR_MFG_PHYSICAL_UNIT_0_AUX_2 to HR_MFG_PHYSICAL_UNIT_14_AUX_2

These registers contain additional information depending on the value of the HR_MFG_PHYSICAL_UNIT_? register.

UNIT	AUX0	AUX1	AUX2
33	LSW of k Float value	MSW of k Float value	UNUSED

Other holding registers

These are some utility Holding Registers.

Address	Name
0x0010	HR_RESET
0x5AA6	HR_TEST

HR_RESET

This write only register is used to issue special command to the SISGEO digital instrument. Two command are currently defined:

Value	Description
0xEE	Reset the unit
0xFF	Power the unit down
0x01	Force sensor supply off (for testing)
0x02	Resume normal power operation

HR_TEST

This register is used to enable different test modes; the following modes are defined:

Value	Description
0x0000	Switch the LED off
0x0001	Switch the LED on
0x??02	Blink the LED when there is a match with the address in the high byte of this command: can be used to detect a message on the MODBUS line directed to a different unit. The value to write in this register is $2 + 256 * \langle \text{address} \rangle$
0x0003	Blink the LED when a message is received: the destination address of the message must either match this unit address or be the broadcast address (that is the address 0).
0x0004	Blink the LED when a message is transmitted

0x0005	Blink the LED when a read cycle on all channels is complete, that is the averages have been completed and another value has been added to the moving averages.
0x0006	Blink the LED at 1Hz rated: the crystal oscillator is used as time base
0x0007	Blink the LED at 1Hz rated: the RC oscillator is used as time base

HR_FACTORY_CONFIG_CHECKSUM0 to HR_FACTORY_CONFIG_CHECKSUM7 and HR_USER_CONFIG_CHECKSUM0 to HR_USER_CONFIG_CHECKSUM7

These read only registers hold a checksum of the Factory and User configurations; the checksum algorithm used is MD5 which yields checksums that are 16 bytes (eight registers long). These registers can be used to buffer configurations on the server: if the checksum has not changed, the configuration has not changed.

Address	Name
0xFFFF0 to 0xFFFF7	HR_FACTORY_CONFIG_CHECKSUM0 to HR_FACTORY_CONFIG_CHECKSUM7
0xFFFF8 to 0xFFFFF	HR_USER_CONFIG_CHECKSUM0 to HR_USER_CONFIG_CHECKSUM7

Note: the checksum is calculated on the whole area of configuration registers, even when some configuration registers are unused. So Factory checksum is calculated on all the registers in range 0x0200 to 0x05FF and User checksum is calculated on all the registers in range 0x1000 to 0x11FF.

4. Legacy mode operation

For a definition of Legacy and User registers, see "User and Legacy configuration holding registers" chapter.

The Holding registers in range 0x0106 to 0x0141 are only used in Legacy mode.

Address	Name
0x0106	HR_LEG_AVERAGE
0x0107 to 0x0141	Calibration parameters

The configuration registers in range 0x0107 to 0x0141 contain the calibration parameters and have the same meaning as in the Legacy SISGEO digital instrument.

When in Legacy mode the HR_LEG_AVERAGE register is used instead of the AVERAGE register for channels 0 and 1: these are the channels normally used to connect the inclinometers or the external instruments.

All other values are taken by the User configuration registers from 0x0148 to 0x01BF.

Offset	Name	Description
+0	Warmup	Used for all channels
+1	Average	Overridden by HR_LEG_AVERAGE for channels 0 and 1
+2	MovingAverage	Used for all channels
+3	Cooldown	Used for all channels
+4	Scale LSW	Used for all channels
+5	Scale MSW	Used for all channels
+7	Model	Used for all channels

The scale parameter in Factory configuration is ignored and a scale factor of 65536 is always used so that fixed point registers

IR_PHYSICAL_INT_CHANNEL_0 to IR_PHYSICAL_INT_CHANNEL_14

always show the integer part of the corresponding floating point register

IR_PHYSICAL_CHANNEL_0 to IR_PHYSICAL_CHANNEL_14

and registers

IR_PHYSICAL_INT_CHANNEL_0+1 to IR_PHYSICAL_INT_CHANNEL_14+1

show the fractional part.

5. Reading a SISGEO digital instrument

This section describes how to read a SISGEO digital instrument step by step. Further details about the MODBUS registers used can be found in the previous sections of this document.

The communication mode used is RTU (Remote Terminal Unit); ASCII mode is not supported. The serial line parameters are: 9600 baud, 8 data bits, 1 stop bit, no parity.

Please, pay attention to the difference between Input and Holding registers.

Reading 32-bit values

The MODBUS standard defines only 16-bit registers; there is no definition in the standard for any 32-bit integer, fixed point or floating-point type.

Since these types are required for the implementation of SISGEO digital instruments, they are simulated by couples of successive registers to be joined to form the 32-bit value.

The following paragraphs describe how this joining is interpreted for the different formats used.

Note: in some cases, the most significant part is contained in the first register, in other cases it is in the second; pay attention to this while implementing your code.

Reading 32-bit integer values from two consecutive registers

To extract a floating-point value from a couple of registers (e.g. register 0x0202 and 0x0203) is as follows:

1. Read the first register as a 16 bit unsigned integer
2. Read the second register as a 16 bit unsigned integer
3. Concatenate the two registers in a 32 bit integer (0x0202 in the most significant 16 bits, 0x0203 in the least significant 16 bits)

The following C/C++ can be used to obtain the translated value:

```
#include "stdint.h"
:
uint16_t msw = ReadRegister(0x0202);
uint16_t lsw = ReadRegister(0x0203);
int32_t i32 = msw;
i32<<=16; // Shift left i32, not msw which is an uint16_t
i32 |= lsw
```

Note the use of the standard types defined in `stdint.h`, preferable to the use of short and long attributes that are machine dependent.

The C# code would be similar.

Reading 16.16 bit values from two consecutive registers (16 bit fixed point values, 16 bit decimal point values)

When values are stored in this format the integer and fractional parts are recorded in two separated 16 bit values.

To extract a floating point value from a couple of registers (e.g. register 0x0202 and 0x0203) proceed as follows:

1. Read the first register as a 16 bit unsigned integer
2. Read the second register as a 16 bit unsigned integer
3. Chain the two registers in a 32 bit integer (0x0202 in the most significant 16 bits, 0x0203 in the least significant 16 bits)
4. Interpret the content of the 32 bit integer as a float.
5. Divide by 65536.0

For example in C or C++:

```
#include "stdint.h"
:
uint16_t msw = ReadRegister(0x0202);
uint16_t lsw = ReadRegister(0x0203);
int32_t i32 = msw;
i32<<=16; // Shift left i32, not msw which is an uint16_t
i32 |= lsw
float value = (float)i32 / 65536.0f;
```

Note the use of the standard types defined in stdint.h, preferable to the use of short and long attributes that are machine dependent.

Reading an IEEE 754 binary 32 floating point value from two consecutive registers

To extract a floating-point value from a couple of registers (e.g. register 0x0202 and 0x0203) proceed as follows:

1. Read the first register as a 16 bit unsigned integer
2. Read the second register as a 16 bit unsigned integer
3. Chain the two registers in a 32 bit integer (0x0202 in the most significant 16 bits, 0x0203 in the least significant 16 bits)
4. Interpret the content of the 32 bit integer as a float.

This is quite easy to accomplish in C and C++.

```
#include "stdint.h"
:
uint16_t msw = ReadRegister(0x0202);
uint16_t lsw = ReadRegister(0x0203);
uint32_t u32 = msw;
u32<<=16; // Shift left u32, not msw which is an uint16_t
u32 |= lsw
float f = *(float*) &u32;
```

Note the use of the standard types defined in `stdint.h`, preferable to the use of short and long attributes that are machine dependent.

Less 'low level' languages usually include libraries to accomplish this, usually by writing the unsigned integers to a stream of byte and reading back the stream of bytes as a float. For example, in C#:

```
uint16 msw = ReadRegister(0x0202);
uint16 lsw = ReadRegister(0x0203)
byte[] bh = BitConverter.GetBytes(msw);
byte[] bl = BitConverter.GetBytes(lsw);
byte[] bf = new byte[4];
bf[0] = bl[0];
bf[1] = bl[1];
bf[2] = bh[0];
bf[3] = bh[1];
return BitConverter.ToSingle(bf, 0);
```

Waiting for SISGEO digital instrument to be ready

In order to reduce power consumption, it is possible to configure a period after the unit is powered up in which the SISGEO digital instrument is sleeping and not available for MODBUS communication.

Shutting down SISGEO digital instrument

At any time, when communication with SISGEO digital instrument is no longer needed, it is possible to switch the instrument off by writing to the HR_RESET Holding register.

Address	Name
0x0010	HR_RESET

The value for shutting down the unit is:

Value	Description
0xFF	Power the unit down

To start communicating with SISGEO digital instrument again, it will be necessary to cycle the power to the unit. Off and on again

Detecting the version of the SISGEO digital instrument

To detect the firmware version of the SISGEO digital instrument, read Holding Register 0x0000 (Version Holding Registers in the “Factory configuration Holding Registers” section).

Address	Name
0x0000	HR_VERSION_FW

The version number should be interpreted in hexadecimal format; the most significant byte (MSB) contains the mayor version, while the least significant byte (LSB) contains the minor version.

This chapter describes reading firmware versions greater or equal to 0x0300.

Older versions are here called Legacy and are described in a separate document.

This Holding Register is not available in some old SISGEO digital instruments, so if this Holding Register is not available it can be assumed that version is less than 0x0300.

The LSB of this Holding Register is not relevant; for example, the current version of SISGEO digital instrument firmware at the time of this writing is 17, so this Holding Register would read 0x0311: 0x11 hexadecimal being 17.

Detecting the operating mode

The SISGEO digital instrument can operate in two different modes:

- Legacy mode
- Normal mode

To detect the operating mode of SISGEO digital instrument, read Holding Register 0x0200 (“Factory configuration Holding Registers” section).

Address	Name
0x0200	HR_MFG_MODE

Depending on the value in this Holding Register:

- 0x5A01: Legacy mode is used; continue with “Detecting Legacy mode configuration” paragraph

- 0x5A02: Normal mode is used; continue with “Detecting Normal mode configuration” paragraph

Before proceeding, it is possible to read the serial number Holding Registers:

Address	Name
0x0202	HR_MFG_SERIAL1
0x0202	HR_MFG_SERIAL2

These must be interpreted as described in the paragraph “HR_MFG_SERIAL1 and HR_MFG_SERIAL2”, and can be used to keep track of the single SISGEO digital instrument.

Detecting Legacy mode configuration

To detect the configuration of SISGEO digital instrument read Input Register 0x0101. Note that this is an Input Register event if it is actually used to hold a configuration value.

Address	Name
0x0102	IR_CONF

The bits in this Input Register define the configuration of the SISGEO digital instrument

BIT	
7-4	Unused
3-2	Unit of measure: 00: k·Sin(α) 01: Degrees (sexagesimal) 10: mm/m 11: Polynomial
1-0	Number of channels (excluding temperature measurement): 01: Single channel 10: Dual channel

When unit is “00: Sin(α)” or “10: mm/m” Holding Register 0x0140 must be read.

Address	Name
0x0140	HR_K_LSW
0x0141	HR_K_MSW

These Holding Registers must be combined as described above to form an IEEE 754 binary 32 floating point value.

- If the unit is “00: k·Sin(α)”, this value is the multiplicative value k.
- If the unit is “10: mm/m”, if this value is 12, it means that the actual unit of measure is inch/feet.

When unit is “11: Polynomial” the Holding Register 0x013B must be read:

Address	Name
0x013B	HR_UNIT

The value of this Holding Register defines the unit of measure for the input channels:

BIT	
7-4	Measure unit for channel 1
3-0	Measure unit for channel 0

According to the following table:

Value	Unit	Value	Unit	Value	Unit
1	mV	2	Bar	3	mBar
4	Atm	5	PSI	6	Pa
7	Kpa	8	MPa	9	mmH ₂ O
10	mH ₂ O	11	InchH ₂ O	12	FeetH ₂ O
13	mmHg	14	cmHg	15	inHg
16	Kg/cm ²	17	Kg/m ²	18	lb/in ²
19	lb/ft ²	20	N/cm ²	21	N/m ²
22	t/m ²	23	t(UK)/ft ²	24	t(USA)/ft ²

Reading input values in Legacy mode as Floating Point

In order to read floating point values in Legacy mode, first read the IR_COUNT Input Register.

Address	Name
0x0100	IR_COUNT

This Input Register reports the number of completed reading and is incremented each time a new average is completed. In this way reading the same values twice can be avoided.

Then read the Input Registers containing the converted values in floating point format.

Address	Name
0x0126	Channel 0 Most Significant Word
0x0127	Channel 0 Least Significant Word
0x0128	Channel 1 Most Significant Word
0x0129	Channel 1 Least Significant Word
0x012A	Temperature Most Significant Word
0x012B	Temperature Least Significant Word

The readings can be converted from couples of 16 bit Input Registers to floating point as described above.

Reading input values in Legacy mode as Fixed Point

In order to read floating point values in Legacy mode, first read the IR_COUNT Input Register.

Address	Name
0x0100	IR_COUNT

This Input Register reports the number of completed reading and is incremented each time a new average is completed. In this way reading the same readings twice can be avoided. Then read the Input Registers containing the converted values in 16.16 bit format as described above.

Address	Name
0x0120	Channel 0 Most Significant Word
0x0121	Channel 0 Least Significant Word
0x0122	Channel 1 Most Significant Word
0x0123	Channel 1 Least Significant Word
0x0124	Temperature Most Significant Word
0x0125	Temperature Least Significant Word

Detecting Normal mode configuration

To detect the configuration of SISGEO digital instrument read Holding Registers 0x0380 to 0x03BB.

Address	Name
0x0380	HR_MFG_PHYSICAL_UNIT_0
0x0381	HR_MFG_PHYSICAL_UNIT_0_AUX_0
0x0382	HR_MFG_PHYSICAL_UNIT_0_AUX_1
0x0383	HR_MFG_PHYSICAL_UNIT_0_AUX_2
:	:
0x0388 to 0x038B	Physical unit description for channel 2
0x03B8 to 0x03BB	Physical unit description for channel 14

There are four Holding Registers for each configurable channel (0 to 14). the addresses of these Holding Registers are:

- HR_MFG_PHYSICAL_UNIT_<channel> 0x0380+4*<channel>
- HR_MFG_PHYSICAL_UNIT_<channel>_AUX_0 0x0380+4*<channel> + 1

- HR_MFG_PHYSICAL_UNIT_<channel>_AUX_1 0x0380+4*<channel> + 2
- HR_MFG_PHYSICAL_UNIT_<channel>_AUX_2 0x0380+4*<channel> + 3

The first Holding Register for each channel (HR_MFG_PHYSICAL_UNIT_<channel>) contains a constant that identifies the unit of measurement as reported in the paragraph "HR_MFG_PHYSICAL_UNIT_0 to HR_MFG_PHYSICAL_UNIT_14".
If the value is 0 the channel is not used and it can be ignored.

The other Holding Registers in each group of four are only relevant for the value $34=k\sin(\alpha)$:
In this case the other Holding Registers contain the value for k.

UNIT	AUX0	AUX1	AUX2
33	LSW of k Float value	MSW of k Float value	UNUSED

Note that the word ordering is reversed (that is, least significant word goes first, compared to the Normal MODBUS convention).

Note: the use of channels 0 to 12 depends on the individual SISGEO digital instrument; the last three channels (12, 13 and 14) are Normally used as follows

CHANNEL	UNIT	Description
12	°C (Celsius)	Temperature inside the instrument (on the electronic board)
13	RH%	Humidity inside the instrument (on the electronic board)
14	Volt	Supply voltage

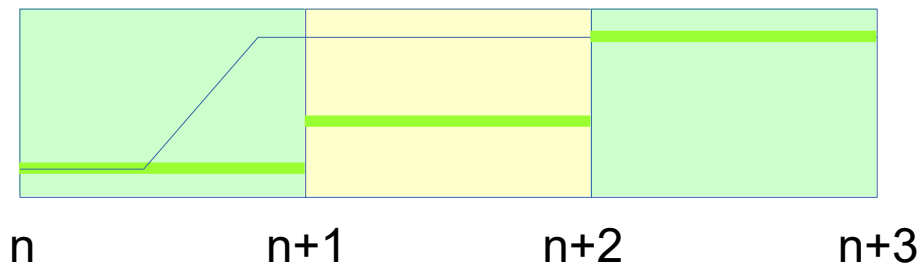
Reading input values in Normal mode as Floating Point

It is possible to read input values in floating point format, reading the Input Registers 0x0200 to 0x021F.

Address	Name
0x200	IR_PHYSICAL_COUNT
0x201	IR_PHYSICAL_CHANGE
0x202	IR_PHYSICAL_CHANNEL_0 most significant word
0x203	IR_PHYSICAL_CHANNEL_0 least significant word
0x204	IR_PHYSICAL_CHANNEL_1 most significant word
0x205	IR_PHYSICAL_CHANNEL_1 least significant word
:	:
0x21E	IR_PHYSICAL_CHANNEL_14 most significant word
0x21F	IR_PHYSICAL_CHANNEL_14 least significant word

When reading a channel, SISGEO digital instrument first performs an average on the value read, then a moving average on the averaged values.

The IR_PHYSICAL_COUNT value is incremented each time an average cycle is completed, while the IR_PHYSICAL_CHANGE value is incremented each time a moving average cycle is completed. So the latter should be used as a settling indicator: wait two increments from the last input change to make sure that the value has settled.



In the above graph, the thin line represents the input value and the green line the moving average. It can be seen that the moving average settles up to two periods after the input value has changed: any change occurred during the period starting at time 'n' is only completely accounted for at time 'n+2'.

The Input Registers starting at 0x202 contain the values read, two Input Registers for channels in 32-bit floating point format; the values can be converted as described above.

Reading input values in Normal mode as Integers

It is possible to read input values in floating point format, reading the Input Registers 0x0220 to 0x024F.

Address	Name
0x220	IR_PHYSICAL_INT_COUNT
0x221	IR_PHYSICAL_INT_CHANGE
0x222	IR_PHYSICAL_INT_CHANNEL_0 most significant word
0x223	IR_PHYSICAL_INT_CHANNEL_0 least significant word
0x224	IR_PHYSICAL_INT_CHANNEL_1 most significant word
0x225	IR_PHYSICAL_INT_CHANNEL_1 least significant word
:	:
0x23E	IR_PHYSICAL_INT_CHANNEL_14 most significant word
0x23f	IR_PHYSICAL_INT_CHANNEL_14 least significant word

The meaning of the Input Registers IR_PHYSICAL_INT_COUNT and IR_PHYSICAL_INT_CHANGE are the same as that of Input Registers IR_PHYSICAL_COUNT and IR_PHYSICAL_CHANGE and the same considerations apply for average and moving average.

The Input Registers starting at 0x222 contain the values read, two Input Registers for channels in 32-bit integer format; the values can be converted as described above.

An external system could just read the first of a couple of Input Registers, in case only 16 bits of resolution are is required.

Some measures would be meaningless if directly transformed into an integer (for example the sine of an angle). For this reason, the configuration registers define multiplicative factors used to obtain the value of IR_PHYSICAL_INT_CHANNEL from the respective IR_PHYSICAL_CHANNEL.

The multiplicative factors, in floating point format, can be read from configuration Holding Registers:

Offset	Name
0x0148+4	Scale LSW for IR_PHYSICAL_INT_CHANNEL0
0x0148+5	Scale MSW for IR_PHYSICAL_INT_CHANNEL0
0x0150+4	Scale LSW for IR_PHYSICAL_INT_CHANNEL1
0x0150+5	Scale MSW for IR_PHYSICAL_INT_CHANNEL1
:	:
0x01B8+4	Scale LSW for IR_PHYSICAL_INT_CHANNEL14
0x01B8+5	Scale MSW for IR_PHYSICAL_INT_CHANNEL14

In general, the values are contained in Holding Registers at addresses 0x0148+4+8*<channel> and 0x0148+5+8*<channel>.

Note that, as for all the configuration registers, word ordering is reversed compared to the Normal MODBUS use, the values can be converted as described above.

For example, if the input value for a sin(alpha) in channel 0 is 0.5 (0x3F00'0000 in IEEE format) and the multiplicative factor is 20000 (0x469C'4000), the expected corresponding integer value would be 10000 (0x0000'2710)

The following values would be read in the Input Registers:

Address	Value
0x202	0x3F00 MSW first, IEEE 754 binary 32 floating point format
0x203	0x0000
0x222	0x0000 MSW first, integer format
0x223	0x2710

And the following in the configuration Holding Registers:

Offset	Name
0x0148+4	0x4000 LSW First IEEE 754 binary 32 floating point format
0x0148+5	0x469C