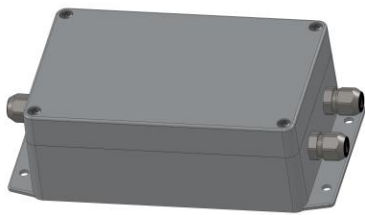


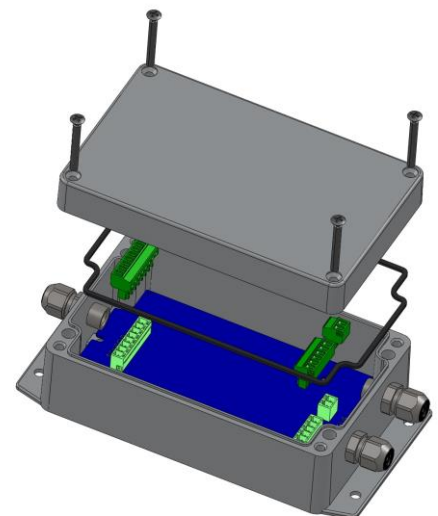
MOBUS – Modbus Adapter



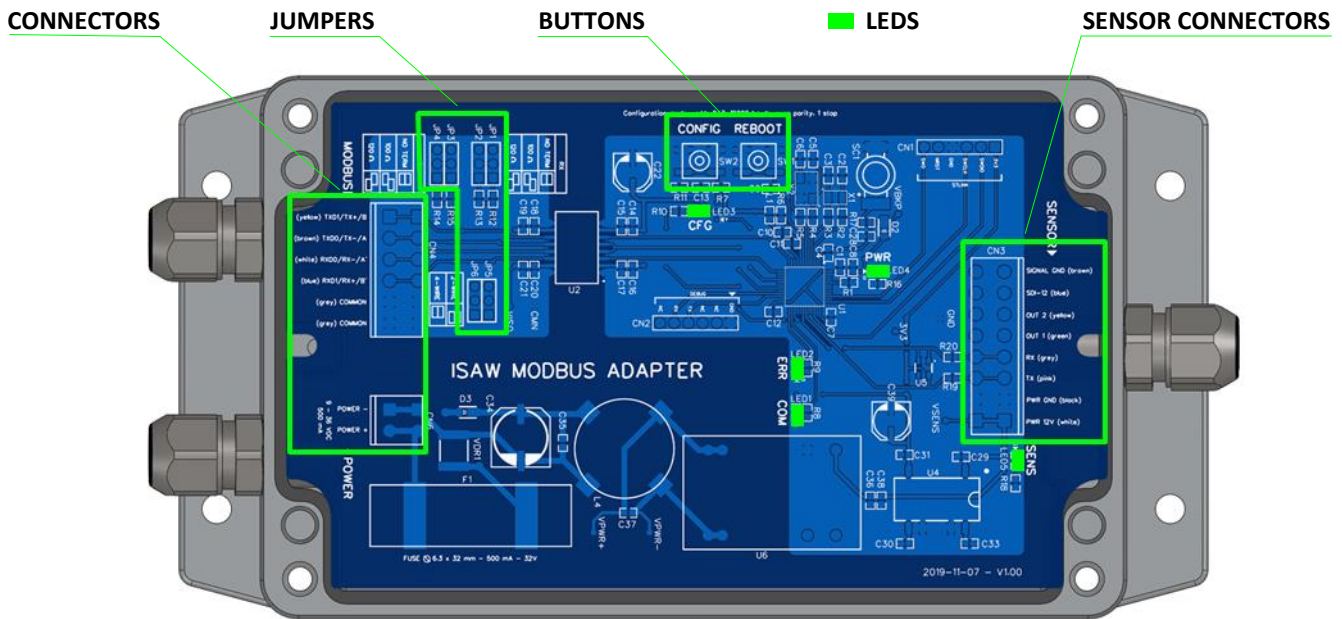
The ISAW Modbus RTU RS485 Adapter (MOBUS) enables the power supply and connection of any ISAW sensor to a Modbus network. MOBUS is the recommended accessory to interconnect the sensor through the open serial Modbus RTU (RS-485) protocol based on a master/slave or client/server architecture. The fieldbus environment is the base level group of digital networks in the hierarchy of plant networks.

CHARACTERISTICS	
Protocol	MODBUS RTU (V1.1b3)
Physical Layer	EIA/TIA-485 (RS485) 2-wire and 4-wire
Galvanic isolation	Power 3kV RMS, Bus 5kV RMS
Unit load	1/8 unit load, up to 256 nodes on the bus
Startup time	1 s
Power supply	9 to 36 VDC (Typ. 100 mA, Max. 500 mA)
Operating temperature	-40°C to 80°C
Implemented function codes	0x04 Read Input Registers 0x03 Read Holding Registers 0x06 Write Single Register 0x10 Write Multiple Registers 0x64 Pass-through 0x08 Diagnostic 0x17 Report Server ID
Configurable Baud rate	9600, 19200, 38400, 57600, 115200, 128000, 256000
Configurable Parity	No, Odd, Even
Configurable Stop Bit	1 or 2
Configurable Address	1 to 247

DEFAULT COMMUNICATION PARAMETERS	
Address	247
Baudrate	19200 bauds
Parity	Even
Stop bits	1 bit
Response timeout	1000 ms



Description



■ BUTTONS

BUTTONS	
REBOOT	Restarts the Modbus adapter with the Holding register's parameters. Note: Switching the power OFF/ON also restarts the Modbus adapter.
CONFIG	Holding the CONFIG button pressed while starting (or restarting) the Modbus adapter starts (or restarts) the adapter with the default communication parameters (cf. previous page). Note: This operation does not change the parameters stored in the Holding register.

■ LEDS

LEDS	
COM	Flashing during a Modbus communication.
ERR	Flashing when a Modbus communication error occurs. Steady when a critical error occurs requiring a restart.
CFG	Flashing when the sensor is in CONFIG mode (started with the CONFIG button pressed).
PWR	Steady when input power OK.
SENS	Steady when output 12 V sensor power OK.

■ CONNECTORS

SENSOR			
#	Name	Color	Description
1	PWR 12V	White	Power output 12VDC – 300mA
2	PWR GND	Black	Power ground
3	TX	Pink	Serial input 3V3
4	RX	Grey	Serial output 3V3
5	OUT 1	Green	Not connected
6	OUT 2	Yellow	Not connected
7	SDI-12	Blue	Not connected
8	Signal GND	Brown	Not connected

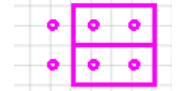
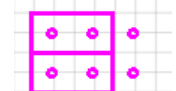
POWER			
#	Name	Color	Description
1	PWR -	Black	Power ground
2	PWR +	Red	Power input 9...36VDC (500mA)

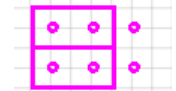
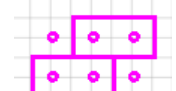
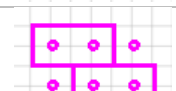
MODBUS 4-WIRE			
#	Name	Color	Description
1	TXD1/TX+/B	Yellow	Output terminal 1, Vb voltage (Vb > Va for binary 1)
2	TXD0/TX-/A	Brown	Output terminal 0, Va voltage (Va > Vb for binary 0)
3	RXD0/RX-/A'	White	Input terminal 0, Va' voltage (Va' > Vb' for binary 0)
4	RXD1/RX+/B'	Blue	Input terminal 1, Vb' voltage (Vb' > Va' for binary 1)
5	COMMON	Grey	Signal ground
6	COMMON	Grey	Signal ground

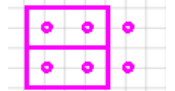
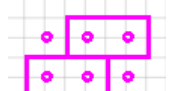
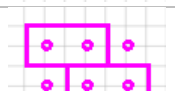
MODBUS 2-WIRE			
#	Name	Color	Description
1	TXD1/TX+/B	Yellow	Transceiver terminal 1, Vb voltage (Vb > Va for binary 1)
2	TXD0/TX-/A	Brown	Transceiver terminal 0, Va voltage (Va > Vb for binary 0)
3	RXD0/RX-/A'	White	Not connected
4	RXD1/RX+/B'	Blue	Not connected
5	COMMON	Grey	Signal ground
6	COMMON	Grey	Signal ground

Note: TXD0-RXD0 and TXD1-RXD1 are connected.

JUMPERS

MODE	
Jumper position	Description
	2-WIRE MODE [default] (TXD0-RXD0 et TXD1-RXD1 are connected)
	4-WIRE MODE

TX TERMINATION	
Jumper position	Description
	NO TERM. [default] No termination resistor on TX pair
	120 Ω Standard 120 Ω termination resistor wired on TX pair
	100 Ω 100 Ω termination resistor wired on TX pair

RX TERMINATION	
Jumper position	Description
	NO TERM. [default] No termination resistor on RX pair
	120 Ω Standard 120 Ω termination resistor wired on RX pair
	100 Ω 100 Ω termination resistor wired on RX pair

Note: If the ISAW Modbus adapter node is the last one on the bus, the bus TX termination must be wired (in 2-WIRE mode, only on one of the two pairs).

Modbus function / Register definition

INPUT Registers

The Input registers contain measurements. The content of these registers is updated each time the sensor sends new measurements to the Modbus adapter.

Modbus function:

- READ INPUT REGISTER (0x04)

FlowCapt FC4, SandFlow SF4 and WindFlow WF4						
Address	Type*	Bytes	Offset	Alias	FC4/SF4	WF4
0x0000	UINT32	4	0	Counter	Flux measurement frame counter	N.A.
0x0002	STRING8	8	2	Unit	Flux measurement unit: "g/m ² /s"	N.A.
0x0006	FLOAT	4	6	Min	Flux measurement minimum	N.A.
0x0008	FLOAT	4	8	Avg	Flux measurement average	N.A.
0x000A	FLOAT	4	10	Max	Flux measurement maximum	N.A.
0x000C	FLOAT	4	12	Std	Flux measurement standard deviation	N.A.
0x000E	STRING8	8	14	Unit	Cumulative flux measurement unit: "g/m ² "	N.A.
0x0012	FLOAT	4	18	Sum	Cumulative flux measurement	N.A.
0x0014	UINT32	4	20	Counter	Wind measurement frame counter	
0x0016	STRING8	8	22	Unit	Wind measurement unit: "km/h"	
0x001A	FLOAT	4	26	Min	Wind measurement minimum	
0x001C	FLOAT	4	28	Avg	Wind measurement average	
0x001E	FLOAT	4	30	Max	Wind measurement maximum	
0x0057	UINT16	2	87	UINT16 Test	Fixed value: 54321 (0xD431)	
0x0058	UINT32	4	88	UNIT32 Test	Fixed value: 1234567890 (0x499602D2)	
0x005A	FLOAT	4	90	FLOAT Test	Fixed value: 3,14159265 (0x40490FDB)	

RainFlow RF4 and HailFlow HF4						
Address	Type*	Bytes	Offset	Alias	RF4	HF4
0x0000	UINT32	4	0	Counter	Rain measurement frame count	Hail measurement frame count
0x0002	STRING8	8	2	Unit	Rain measurement unit: "mm/h"	Hail measurement unit: "hit/h"
0x0006	FLOAT	4	6	Min	Rain measurement minimum	Hail measurement minimum
0x0008	FLOAT	4	8	Avg	Rain measurement average	Hail measurement average
0x000A	FLOAT	4	10	Max	Rain measurement maximum	Hail measurement maximum
0x000C	FLOAT	4	12	Std	Rain measurement stand. deviation	Hail measurement stand. deviation
0x000E	STRING8	8	14	Unit	Cum. rain measurement unit: "mm"	Cum. hail measurement unit: "hit"
0x0012	FLOAT	4	18	Sum	Cumulative rain measurement	Cumulative hail measurement
0x0020	UINT32	4	32	Counter	Drop measurement frame count	Hailstone measurement frame count
0x0022	STRING8	8	34	Unit	Drop count unit: "hit"	Hailstone count unit: "hit"
0x0026	UINT32	4	38	Hit count	Drop count	Hailstone count
0x0028	STRING8	8	40	Unit	Drop distribution unit: "%"	Hailstone distribution unit: "%"
0x002C	UINT16	2	44	Classe 1	Drop distribution of the class 1	Hailstone distribution of the class 1
0x002D	UINT16	2	45	Classe 2	Drop distribution of the class 2	Hailstone distribution of the class 2
0x002E	UINT16	2	46	Classe 3	Drop distribution of the class 3	Hailstone distribution of the class 3
0x002F	UINT16	2	47	Classe 4	Drop distribution of the class 4	Hailstone distribution of the class 4

RainFlow RF4 and HailFlow HF4						
Address	Type*	Bytes	Offset	Alias	RF4	HF4
0x0030	UINT16	2	48	Classe 5	Drop distribution of the class 5	Hailstone distribution of the class 5
0x0031	UINT16	2	49	Classe 6	Drop distribution of the class 6	Hailstone distribution of the class 6
0x0032	UINT16	2	50	Classe 7	Drop distribution of the class 7	Hailstone distribution of the class 7
0x0033	UINT16	2	51	Classe 8	Drop distribution of the class 8	Hailstone distribution of the class 8
0x0034	UINT16	2	52	Classe 9	Drop distribution of the class 9	Hailstone distribution of the class 9
0x0035	UINT16	2	53	Classe 10	Drop distribution of the class 10	Hailstone distribution of the class 10
0x0036	UINT16	2	54	Classe 11	Drop distribution of the class 11	Hailstone distribution of the class 11
0x0037	UINT16	2	55	Classe 12	Drop distribution of the class 12	Hailstone distribution of the class 12
0x0038	UINT16	2	56	Classe 13	Drop distribution of the class 13	Hailstone distribution of the class 13
0x0039	UINT16	2	57	Classe 14	Drop distribution of the class 14	Hailstone distribution of the class 14
0x003A	UINT16	2	58	Classe 15	Drop distribution of the class 15	Hailstone distribution of the class 15
0x003B	UINT16	2	59	Classe 16	Drop distribution of the class 16	N.A.
0x003C	UINT16	2	60	Classe 17	Drop distribution of the class 17	N.A.
0x003D	UINT16	2	61	Classe 18	Drop distribution of the class 18	N.A.
0x003E	UINT16	2	62	Classe 19	Drop distribution of the class 19	N.A.
0x003F	UINT16	2	63	Classe 20	Drop distribution of the class 20	N.A.
0x0040	UINT16	2	64	Classe 21	Drop distribution of the class 21	N.A.
0x0041	UINT16	2	65	Classe 22	Drop distribution of the class 22	N.A.
0x0042	UINT16	2	66	Classe 23	Drop distribution of the class 23	N.A.
0x0043	UINT16	2	67	Classe 24	Drop distribution of the class 24	N.A.
0x0044	UINT16	2	68	Classe 25	Drop distribution of the class 25	N.A.
0x0045	UINT16	2	69	Classe 26	Drop distribution of the class 26	N.A.
0x0046	UINT16	2	70	Classe 27	Drop distribution of the class 27	N.A.
0x0047	UINT32	4	71	Counter	Hail measurement frame counter	N.A.
0x0049	STRING8	8	73	Unit	Hail measurement unit: "hit"	N.A.
0x004D	UINT32	4	77	Hit count	Hail measurement	N.A.
0x004F	STRING8	8	79	Unit	Hail measurement rate unit: "hit/s"	N.A.
0x0053	FLOAT	4	83	Mean Rate	Hail measurements mean rate	N.A.
0x0055	FLOAT	4	85	Max Rate	Hail measurement max rate	N.A.
0x0057	UINT16	2	87	UINT16 Test	Fixed Value: 54321 (0xD431)	
0x0058	UINT32	4	88	UNIT32 Test	Fixed value: 1234567890 (0x499602D2)	
0x005A	FLOAT	4	90	FLOAT Test	Fixed value: 3,14159265 (0x40490FDB)	

* **Note:** String are zero-padded.

HOLDING Registers

Holding registers are mainly used to configure the Modbus adapter communication.

Note: Restart the Modbus adapter after changing the configuration.

Modbus functions:

- READ HOLDING REGISTERS (0x03)
- WRITE SINGLE REGISTER (0x06)
- WRITE MULTIPLE REGISTERS (0x10)

Address	Type	Bytes	Offset	Name	Values
0x0000	UINT32	4	0	Serial speed	9600, 19200 [default], 38400, 57600, 115200, 128000, 256000
0x0002	UINT16	2	2	Parity	0: No parity, 1: Even [default], 2: Odd
0x0003	UINT16	2	3	Stop Bit	1 [default] or 2 (if no parity)
0x0004	UINT16	2	4	Device address	1 to 247 [default]
0x0005	UINT16	2	5	Response timeout (ms)	Default: 1000

Total bytes: 12
Nb. REG: 6

DATA TYPE Format

- UINT16 (Big Endian)

High Byte	Low Byte
-----------	----------

- UINT32 (Big Endian)

High Byte			Low Byte
-----------	--	--	----------

- FLOAT (IEEE-754)

SEEEEEEE	EMMMMMMMM	MMMMMMMMMM	MMMMMMMMMM
----------	-----------	------------	------------

(S: Sign, E: Exponent, M: Mantissa)

- RAW

Char 1	Char 2	Char 3	...
--------	--------	--------	-----

EXAMPLES:

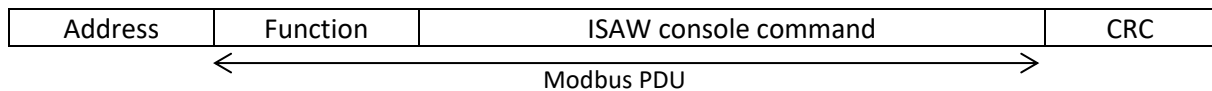
	Decimal	Hexadecimal	Register N	Register N+1
■ UINT16 (Big Endian)	54321	0xD431	0xD431	
■ UINT32 (Big Endian)	1234567890	0x499602D2	0x02D2	0x4996
■ FLOAT (IEEE-754)	3.14159265	0x40490FDB	0x0FDB	0x4049
■ RAW	"hit"	0x68697400	0x6869	0x7400

Modbus function PASS-THROUGH (0x64)

This user-defined Modbus function allows to send an ISAW command through the Modbus (see User Guide – Serial communication for more information about the ISAW console commands). For example, use this function with the “set” or “get” ISAW command to access the sensor’s configuration.

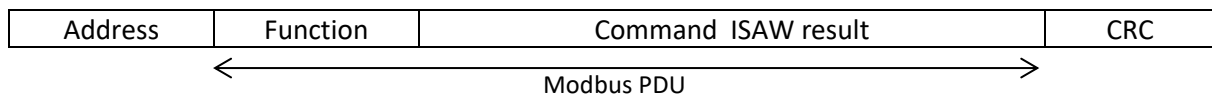
Note: The use of this function is limited by the request and answer lengths.

Request



Field	Size (bytes)	Description
Address	1	Device address (target)
Function	1	0x64
ISAW Command	N	ISAW command as ASCII string
CRC	2	CRC16

Response



Field	Size (bytes)	Description
Address	1	Device address (same as request)
Function	1	0x64
Result	N	ISAW command result as ASCII string
CRC	2	CRC16

Note: The response timeout of this command must be > 2 seconds to allow sensor wake-up.

