MODBUS

This device use MODBUS RTU protocol and it is a “slave”.

What is MODBUS?

It is a application-layer messaging protocol. It provides client/server communication between devices connected on networks.

Devices with MODBUS protocol use 8, N, 1 data format: 8 data bits, no parity, 1 stop bit.
Communication speed rate (baud) can be set from devices communication menu directly.

MODBUS transactions are related always to the “master”, that manage the line and a “slave” device per time (except for broadcast messages).

Each “slave” device is univoquely identified by an **address**.

First character of the message always contains the “slave” device’s numeric address.

Permitted addresses are from 1 to 255. 0 is used only for broadcast messagges, directed to all “slave” devices at the same time.

Second character of the message contains the master’s request. The “slave” device replies with same character to mean the request has been executed.

Frequently used requests are :

Function	Description
01	Read Coil Status
02	Read Input Status
03	Read Holding Registers
04	Read Input registers
05	Force Single Coil
06	Preset Single register
07	Read Status

Last two character of the message contains Cyclic Redundancy Check obtained by CRC16 algorithm.

MODBUS FUNCTIONS

Read Output Status (01)

The function asks the ON / OFF of binary logic variables.
Broadcast is not allowed.
Request
In addition to the “slave” address and the function code (01), message contains starting address on two bytes and the number of bits to be read also on two bytes. The address numbering starts from zero (bit1 = 0) for the MODBUS.

Example: Request to read from the slave's 17-bit 04-015.

ADDR	FUNC	DATA start Addr HI	DATA start Addr LO	DATA bit # HI	DATA bit # LO	CRC HI	CRC LO
11	01	00	03	00	0C	CE	9F

Reply

In addition to the “slave” address and the function code (01), message contains the number of data bytes and characters containing the data.
Data are packed so that a byte represent an 8 bit status, less significant bit of the first byte contains the bit corresponding to the starting Address and so on.
If the number of bits to be read is not multiple of 8, the last character is completed with zeros in the most significant bits.

Example: Reply to the previous request.

ADDR	FUNC	DATA byte count	DATA bit 04..11	DATA bit 12..15	CRC HI	CRC LO
11	01	02	CD	0B	6D	68

Read Output Registers (03)

This function allows to request value of 16-bit (word) registers containing numeric variables.
In addition to the “slave” address and the function code (03), the message contains the starting address on two bytes and the number of words to be read also on two bytes. The maximum number of words that can be read is 125.

Example: : Request to read from slave 25 of registers from 4069 to 40071.

ADDR	FUNC	DATA start Addr HI	DATA start Addr LO	DATA bit # HI	DATA bit # LO	CRC HI	CRC LO
19	03	00	44	00	03	46	06

Reply
In addition to the “slave” and the function code (03), message contains a character that contains the number of data bytes and characters containing the data.
The registers require two bytes each, the first of which contains the most significant part.

Example: Reply to the previous request.

ADDR	FUNC	DATA byte count	DATA byte 69 HI	DATA byte 69 LO	DATA byte 70 HI	DATA byte 70 LO	DATA byte 71 HI	DATA byte 71 LO	CRC HI	CRC LO
19	03	06	02	2B	00	00	00	64	AF	7A

Force Single Coil (05)

This function allows to force a single binary variable state ON or OFF.
In addition to the "slave" address and the function code (05), the message contains the address of the variable to force two bytes and two characters of which the first is set to FF hex (255) to force ON state and 00 hex to force OFF state, the second is set to zero in every case.

Example: Request to force ON on "slave" 47 bit 4.

ADDR	FUNC	DATA bit HI	DATA bit LO	DATA ON/OFF	DATA (Zero)	CRC HI	CRC LO
2F	05	00	03	FF	00	7A	74

Reply
Reply consists in reading setpoint status modification. **See Read Status (07)**

Preset Single Register (06)

This function allows to set a 16 bit single register value.
In addition to the "slave" address and the function code (06) the message contains the address of the variable on two byte and the value to be assigned to.

Example: Request to force 928 on "slave" 35 address 26.

ADDR	FUNC	DATA bit HI	DATA bit LO	DATA WORD HI	DATA WORD LO	CRC HI	CRC LO
23	06	00	19	03	A0	5E	07

Replay

Reply consists in reading setpoint status modification. **See Read Status (07)**

Read Status (07)

This function allows to read status an 8 bit message predetermined with a compact message.
Example: Request on "slave" 25 status.

ADDR	FUNC	CRC HI	CRC LO
19	07	5E	07

Replay
In addition to the "slave" address and the function code (07) the message contains a character with the status bits.

ADDR	FUNC	Status_send	CRC HI	CRC LO
2F	05	00	7A	74

Dove status-send:

0 setpoint changed succesfully
1 wait setpoint changing
2 setpoint changing error

ERROR MANAGEMENT

ADDR	FUNC	DATA exept. code	CRC HI	CRC LO
0A	81	02	7A	74

Exceptional codes

CODE	NAME	DESCRIPTION
01	ILLEGAL FORMAT	Uncorret format message
02	ILLEGAL DATA ADDRESS	Address referred is not allowed on the “slave” .
03	ILLEGAL DATA VALUE	Uncorrect function
04	CRC ERROR	CRC checksum error

ADDRESS LIST VALUE:

Address	Numero Registri	Format	Proprietà	Function	Description
Lettura Valori Attuali					
40002	2	Int16	R	03	Channel1 reading without decimal point
40004	2	Int16	R	03	Current measure dividing factor. Values: 1, 10, 100, 1000
40006	2	Int16	R	03	Channel2 reading without decimal point
40008	2	Int16	R	03	Current measure dividing factor. Values: 1, 10, 100, 1000
40010	2	Int16	R	03	Channel3 reading without decimal point
40012	2	Int16	R	03	Current measure dividing factor. Values: 1, 10, 100, 1000
40014	2	Int16	R	03	Channel4 reading without decimal point
40016	2	Int16	R	03	Current measure dividing factor. Values: 1, 10, 100, 1000
40018	2	Int16	R	03	Channel5 reading without decimal point
40020	2	Int16	R	03	Current measure dividing factor. Values: 1, 10, 100, 1000
Allarmi					
00	1Bit	Bit	R	01	Ch1 Aa alarm 1:On 0:Off
01	1Bit	Bit	R	01	Ch1 Ab alarm 1:On 0:Off
02	1Bit	Bit	R	01	Ch1 Ad alarm 1:On 0:Off
03	1Bit	Bit	R	01	Ch1 Ar alarm 1:On 0:Off

04	1Bit	Bit	R	01	Ch2 Aa alarm 1:On 0:Off
05	1Bit	Bit	R	01	Ch2 Ab alarm 1:On 0:Off
06	1Bit	Bit	R	01	Ch2 Ad alarm 1:On 0:Off
07	1Bit	Bit	R	01	Ch2 Ar alarm 1:On 0:Off
08	1Bit	Bit	R	01	Ch3 Aa alarm 1:On 0:Off
09	1Bit	Bit	R	01	Ch3 Ab alarm 1:On 0:Off
10	1Bit	Bit	R	01	Ch3 Ad alarm 1:On 0:Off
11	1Bit	Bit	R	01	Ch3 Ar alarm 1:On 0:Off
12	1Bit	Bit	R	01	Ch4 Aa alarm 1:On 0:Off
13	1Bit	Bit	R	01	Ch4 Ab alarm 1:On 0:Off
14	1Bit	Bit	R	01	Ch4 Ad alarm 1:On 0:Off
15	1Bit	Bit	R	01	Ch4 Ar alarm 1:On 0:Off
16	1Bit	Bit	R	01	Ch5 Aa alarm 1:On 0:Off
17	1Bit	Bit	R	01	Ch5 Ab alarm 1:On 0:Off
18	1Bit	Bit	R	01	Ch5 Ad alarm 1:On 0:Off
19	1Bit	Bit	R	01	Ch5 Ar alarm 1:On 0:Off
20	1Bit	Bit	R	01	Flow Alarm 1:On 0:Off
21	1Bit	Bit	R	01	Init System 1:On 0:Off
22	1Bit	Bit	R	01	Temperature Alarm 1:On 0:Off
Output					
32	1Bit	Bit	R	01	Ch1 Da digital 1:On 0:Off
33	1Bit	Bit	R	01	Ch1 Db digital 1:On 0:Off
34	1Bit	Bit	R	01	Ch2 Da digital 1:On 0:Off
35	1Bit	Bit	R	01	Ch2 Db digital 1:On 0:Off
36	1Bit	Bit	R	01	Ch3 Da digital 1:On 0:Off
37	1Bit	Bit	R	01	Ch3 Db digital 1:On 0:Off
38	1Bit	Bit	R	01	Ch4 Da digital 1:On 0:Off
39	1Bit	Bit	R	01	Ch4 Db digital 1:On 0:Off
40	1Bit	Bit	R	01	Ch5 Da digital 1:On 0:Off
41	1Bit	Bit	R	01	Ch5 Db digital 1:On 0:Off
42	1Bit	Bit	R	01	Probe Clean – Clean Time 1:On 0:Off
43	1Bit	Bit	R	01	Probe Clean – Restore Time 1:On 0:Off
Input					
24	1Bit	Bit	R	01	standby 1:On 0:Off
25	1Bit	Bit	R	01	Manual Mode 1:On 0:Off
26	1Bit	Bit	R	01	Level 1 1:On 0:Off
27	1Bit	Bit	R	01	Level 2 1:On 0:Off
28	1Bit	Bit	R	01	Level 3 1:On 0:Off
29	1Bit	Bit	R	01	Level 4 1:On 0:Off
30	1Bit	Bit	R	01	Level 5 1:On 0:Off
Current DateTime					
40044	2	Int16	R		MOUNTH|DAY
40046	2	Int16	R		HOUR | YEAR
40048	2	Int16	R		00|MINUTES
Temperature					
40052	2	Int16	R		Misura Temperatura
mV Probe					

40264	2	Int16	R		mV Probe Ch1
40266	2	Int16	R		mV Probe Ch2
40268	2	Int16	R		mV Probe Ch3
40270	2	Int16	R		mV Probe Ch4
40272	2	Int16	R		mV Probe Ch5