

EUROPROT +

Modbus protocol



Document ID: PP-13-20173
Budapest, February 2011

Version	Date	modification	done
Version 1.0	01/02/2011	first release	Protecta
Version 2.0	22/11/2013	new functions (read coil)	Szaniszló

CONTENT

1	Preface.....	4
2	Implementation overview	4
3	Event channels	5
4	Control operation	6
5	Memory map.....	7

1 Preface

Implementation guideline and general description of the vendor specific memory map

2 Implementation overview

The EuroProt+ Modbus implementation supported operations:

- function code 01 is used to read one or multiple coil values
- function code 03 is used to read one or multiple registers.
- function code 06 is used to write single holding register.
- function code 16 is used to write multiple register. Only the time set function allows write multiple registers.

No other Modbus function code implemented currently.

The implementation does not offer any way to access the I/O signals directly, because this is not a remote I/O device. The data from the EuroProt+ to master direction flown via event channels. The event channels are time stamped and buffered. Reading the sequence of events in this way the possible loss of fast data change is eliminated.

There are registers for accessing the event channel values only. These values are 32 bit representations of the event channels and must be interpreted according to the specific event channel data type.

Boolean type event channel values are accessible as coils also.

The control operation from the master to the EuroProt+ using control channels. The EuroProt+ system has sophisticated return codes to inform the client (master) about the result of the operation.

3 Event channels

The event channels are used to access internal status values from EP+. Every change of the event channels are buffered and time stamped.

An event channel can contain boolean type data, enumerated integer status, 32 bit floating point type measurand and 32 bit unsigned integer type counters. The type of the current channel are identified by the channel identification register.

The event time can be accessed in two ways. One is to get the time read a 32 bit unsigned integer number to determine the unix epoch seconds in UTC and the next 32 bit unsigned integer to determine the elapsed microseconds within the second.

The other way is to read the years, months, days, hours, minutes, seconds, milliseconds registers. These registers stores the event time in localtime format. (Please check the time zone settings on the device to get the right local time)

The quality register provides information about the time accuracy and test mode.

The number of event channels and the meaning of the channels are configuration specific.

Every single event channel has the following structure of 16 bit registers:

Address offset	Register name	Description
0	Event channel number	Range: 0..255
1	Channel data type identification	0: this channel is not used 1: boolean 2: enumerated type, range: 0..n 3: 32 bit integer 4: not used 5: 32 bit floating point number 6: 32 bit unsigned integer
2	32 bit current value of the channel	Interpret these bits based on type of the data
3		
4	32 bit last change time seconds in UTC	Elapsed seconds since unix epoch (1970-01-01)
5		
6	32 bit last change time microseconds in UTC	Range 0..999999
7		
8	Last change time years in localtime	Range: 2010-....
9	Last change time months in localtime	Range: 1..12
10	Last change time days in localtime	Range: 1..31
11	Last change time hours in localtime	Range: 0..23
12	Last change time minutes in localtime	Range: 0..59
13	Last change time seconds in localtime	Range 0..59 (can be 60 if leap second inserted)
14	Last change time milliseconds in localtime	Range 0..999
15	Current quality of the event (see IEC61850-7-3 clause 6.2 for detailed explanation)	All bits are cleared means normal event Bit0: Bit1: Test mode event Bit9: Measurand out of range Bit10: Measurand overflown Bit11: Invalid value Bit14: Event reason is periodic update Bit15: Quality changed only, no value changed

This structure describes the current values of the channels and the same structure is used to read the sequence of events.

4 Control operation

Depending on the configuration, the EuroProt+ system can have different control channels for different purposes. A control channel can be mapped to a circuit breaker, disconnecter or a process control. Every control channel has a range which is acceptable. A circuit breaker can have on and off commands only, but the mode of the device can be set to 5 different stage.

The EuroProt+ system always provide a backward channel about the result of the last control operation. It is recommended to poll the result channel after a control execution to get information about the result of the control.

Only the function code 06 is suitable to initiate a control operation. Writing multiple registers in control register area is not allowed and generates an exception.

5 Memory map

Register Address	Data	R/W
0..15	New event structure. The actual event structure can be found here if the previous event is acknowledged and the new event flag is set.	R
16	Acknowledge the new event. Writing hex value 0x5AA5 here means the new event is processed by the client and the next event in the sequence can be shown. All other value written here causes exception. Reading this register: hex value 0x5AA5 means the EuroProt+ still hasn't processed the acknowledgement from the client (master). 0xA8D3 indicates that currently there is no event in the buffer and the EuroProt+ waiting for the new events. 0x12ED shows there are new events in the new event structure. The event is kept until the client (master) acknowledges it.	R/W
17-19	Gap	
20-21	Current device time unix epoch in UTC It is strongly recommended to read register 20-23 together.	R
22-23	Current device time microseconds within the seconds	R
24	Current device time years in localtime	R
25	Current device time months in localtime	R
26	Current device time days in localtime	R
27	Current device time hours in localtime	R
28	Current device time minutes in localtime	R
29	Current device time seconds in localtime	R
30	Current device time milliseconds in localtime It is strongly recommended to read register 24-30 together	R
31-34	Reserved for time set in UTC	W
35-41	Reserved for time set in localtime	W
42-63	Gap	
64	First control channel. Write: the control value. If the written value is out of the range an exception is replied. Read: Control channel last operation result code. The result of the control can be polled here. The value is an integer number. After the control channel operation this value immediately set to 0. After a short time the result code set. It means any non/zero value here indicates the result of the command. Possible results are: -1 (or 0xFFFF): Operation was successful 2: Operation blocked by the switching hierarchy (remote op. not allowed) 4: Invalid position. The operation not causes the excepted result. 5: Position reached. The output is already in the requested state 7: Step limit. The controller reached the upper or lower position 8: Blocked by mode. The operation is inhibited by device mode. 9: Blocked by process. The operation is inhibited by the process. 10: Blocked by interlock. The operation is inhibited by the interlock logic. 11: Blocked by synchrocheck. Circuit breakers "on" command only! 100: Control value out of the range. The written value is not allowed.	R/W
65-127	Additional control channels. The number of control channels and the device behavior is configuration dependent.	R/W
128-255	Gap	
256-271	First event channel structure (16 registers, see the event channel def.)	R
272-287	Second event channel structure	R
288..4295	Following event channel structures, configuration dependent	R
4296-4297	First event channel value (32 bit, 2 registers)	R
4298-4299	Second event channel value (32 bit, 2 registers)	R
4300-4807	Following event channel 32 bit values	R
4808-5064	Boolean type event values (as single coils)	R