Application note AN485

Task

Establishing a Modbus network

Description

Setting up the unit and building an RS485 network with the proper terminating components

.

Unipower

For this task the Unipower HPL500MB is used

Setup

HPL500MB is equipped with a protocol stack to handle Modbus communication. Setting up HPL500MB for this task consist of selecting an address and the baud rate with which the Modbus network is operating.

Address (Parameter P10) must be set to an address not used by any unit in the network; Each unit must have a unique address.

Baud rate (Parameter P11) must be set according to the baud rate for the Modbus network to which the unit will be connected. Default is 19.2 kbaud, but available is also 9.6 kbaud.

Mode of operation

HPL500MB acts as a slave in the Modbus network and will answer requests send to the address set up in Parameter P10. All other communication will be ignored. For a complete list of requests (commands) please refer to the technical information in appendix 1: Modbus Telegram.

Connection

The two schematics below show how the HPL500MB may be connected to either a SCADA system (or any other industrial system) or to the Unipower Modbus Remote Terminal Unit HPL500H.

The complexity of the wiring depends on the length of the cable and the amount of noise in the environment. Following the EIA485 standard as shown in figure 1 requires a number of resistors and extra earth connections. These precautions should be taken if operating with long cables and/or in a noisy environment.

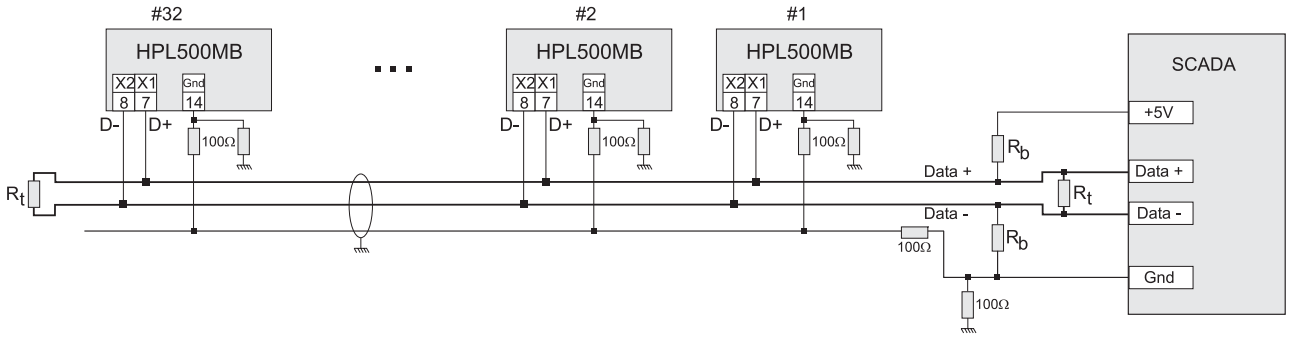

Figure 1. Industrial Modbus network according to EIA485

Terminating resistors (R_t) and bias resistors (R_b) may be selected depending on the number of HPL500MB in the network. Experienced users of RS485/Modbus networks will have hands-on knowledge on these matters, so no solution will be presented here.

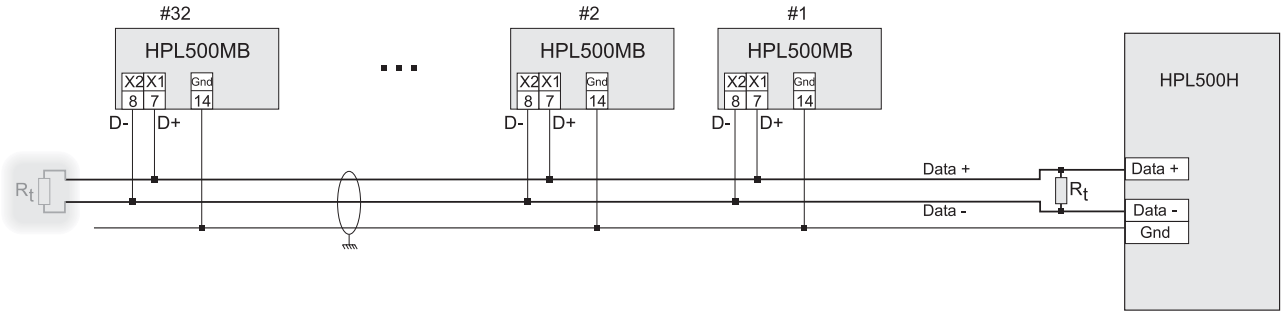

Figure 2. MODbus network with Remote Terminal Unit HPL500H

When using the Unipower Remote Terminal Unit HPL500H the Modbus network may be wired as shown in figure 2. In comparison with the EIA485 standard the ground resistors, bias resistors and some earth connections are omitted. Also the terminating resistor at one end may be omitted (shaded resistor to the left) if the cable length is short, but if telegrams are lost, some of the measures may need to be taken.

The value of the terminating resistor depends on the number of HPL500MB in the network. This is due to the input impedance (R_i) and built-in bias resistors (R_b) in each HPL500MB – see figure 3. The goal is to maintain a voltage between the data lines of at least 200mV in the idle state. Connecting several HPL500MB to the network results in connecting these impedances in parallel and thus changing mainly the bias current. Using the values in table 1 will compensate for the increasing current and produce a balanced network with proper voltage levels.

Please note that the values for the terminating resistor (R_t) assume the use of only one resistor. If both resistors are used, the value must be doubled.

# HPL500MB	Terminating resistor (R_t) (Using only one)
1-2	3.3kΩ
3-5	1.2kΩ
6-9	560Ω
10-14	330Ω
15-32	220Ω

Table 1. Terminating resistor values

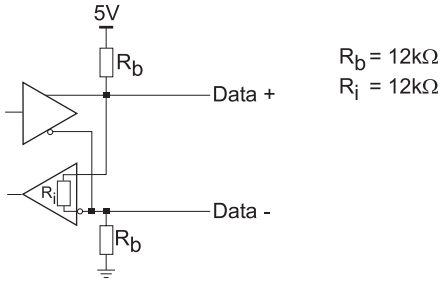

Figure 3. HPL500 input impedance and built-in bias resistors

Modbus telegram for Load monitor HPL500-MB

HPL500-MB uses standard Modbus RTU-protocol, so please refer to the document ”Modbus Application Protocol Specification” for details on the communication. In the following it will be explained how variables and parameters are read and stored in the HPL500. The explanation is limited to showing the used function code and sub parameters. Thus a description of how to construct a telegram is not included. For this, please refer to the official document mentioned earlier.

1. Read measurement(s)

The measurements are called with the Modbus telegram *Read_Input_Register (function code 0x04)* with start address given with table 1 and number of registers as required.

Start address	Measurement	Format
0	kW%	0.1 %
1	kW	0.01 kW
2	Min peak value	%
3	Max peak value	%

Table 1. Start addresses for measurements

Number of registers (Measurements) determines how many values are returned starting with the requested (Start Address).

”Format” in the table explains how the returned values must be interpreted. If for instance a value of 500 is returned for actual measurement (kW%) the measurement was 50.0%.

Example:
A telegram is send with start address 0 and number of registers 2.
HPL500 returns <actual measurement in kW%> & <actual measurement in kW>.

2. Read parameters

The Parameters are requested with the Modbus telegram *Read_Holding_Register (function code 0x03)* with start address given according to table 2 and number of registers as required.

Start address	Parameter	Format
0	Start timer	0.1 s
1	Reaction timer 1	0.1 s
2	Reaction timer 2	0.1 s
3	Set point 1	%
4	Set point 2	%
5	Current range 1	Index – see table 2.1
6	Current range 2	Index – see table 2.1
7	Voltage range	Index – see table 2.2
8	Relay polarity 1	0 = n.inv, 1 = inv
9	Relay polarity 2	0 = n.inv, 1 = inv
10	Limit 1 (Type)	0 = Off, 1 = min, 2 = max
11	Limit 2 (Type)	0 = Off, 1 = min, 2 = max
12	Auto Shut Down	0 = Off, 1 = On
13	Hysteresis 1	0 = Off, otherwise %
14	Hysteresis 2	0 = Off, otherwise %
15	Damping filter	0 = Off, 1 = On
16	S2 function	0 = nor, 1 = SPC
17	Motor efficiency	%
18	Rated shaft power	0.01 kW

Table 2. Start addresses for parameters

Number of registers (Measurements) determines how many values are returned starting with the requested (Start Address).

”Format” in the table explains how the returned values must be interpreted. If for instance a value of 50 is returned for start timer the start timer is 5.0 seconds.

Example:
A telegram is send with start address 0 and number of registers 3.
HPL500 returns <Start timer>, <Reaction timer 1> & <Reaction timer 2>.

2.1 Table for current ranges

Index	Internal	Index	External CT
0	0.5A	8	50/5A
1	1A	9	75/5A
2	2.5A	10	100/5A
3	5A	11	125/5A
4	10A	12	150/5A
5	20A	13	200/5A
6	30A	14	250/5A
7	40A	15	300/5A
		16	400/5A
		17	500/5A
		18	600/5A

Table 2.1 Index for current ranges

2.2 Table for voltage range

Index	Single phase	Index	3 phase
0	1x100V	10	3x100V
1	1x110V	11	3x110V
2	1x120V	12	3x120V
3	1x200V	13	3x200V
4	1x208V	14	3x208V
5	1x220V	15	3x220V
6	1x230V	16	3x230V
7	1x240V	17	3x240V
8	1x380V	18	3x380V
9	1x400V	19	3x400V
		20	3x415V
		21	3x440V
		22	3x460V
		23	3x480V
		24	3x500V
		25	3x525V
		26	3x575V

Table 2.2 Index for voltage range

3. Write parameters

Parameters are stored (written) in HPL500-MB with the Modbus telegram *Write_Multiple_Registers (function code 0x10)* with start address given by table 2 and number of parameters as required. Please note that only the first 5 parameters (0...4) can be changed. The remaining are read-only. Trying to write an invalid parameter or a read only parameter will return an ”exception code 03”.

Example:
The two reaction timers must be changed.
Send a telegram with start address 1 and number of registers 2 followed by the new value for reaction timer 1 and the new value for reaction timer 2 (For further details, please refer to the official document mentioned in the beginning of this document).

4. Read status

Status bits in HPL500-MB are as shown in table 3. All may be called with the Modbus telegram *Read_Coils (function code 0x01)* with start address given by table 3 and number of coils as required.

Start address	Status bit
0	Alarm status for limit 1
1	Alarm status for limit 2
2	Relay 1 position
3	Relay 2 position
4	Start timer active
5	Reaction timer 1 active
6	Reaction timer 2 active
7	Range select (0 = range 1, 1 = range 2)

Table 3. Start addresses for status bits

Example:
Alarm status for the two limits is required.
Send telegram with start address 0 and number of coils 2.
The HPL500 returns a byte with the two least significant bits corresponding to the alarm status 1 and 2. Bit 0 = alarm status for limit 1 and bit 1 = alarm status for limit 2. Alarm active is shown as 1. The remaining bits (bit 2 -> bit 7) = 0.

5. Acquire diagnostics

The mandatory diagnostics values can be acquired with the Modbus telegram *Diagnostics (function code 0x08)* and the following sub codes.

Function code	Sub code (HEX)	Description
08	00	Return Query Data
	0A	Clear counters and diagnostics register
	0B	Return Bus Message Count
	0C	Return Bus Communication Error count
	0D	Return Exception Error count
	0E	Return Slave Message count
	0F	Return Slave No Response count
	12	Return Bus Character Overrun count

Example:
The number of communication errors is required:
Send telegram with function code 08 and sub code 0C.