

## Modbus RTU/TCP Specifications

for device PLU-K-16.16

### Supported Modbus types:

3 = Holding Register

4 = Input Register

### Supportet Modbus functions:

3 = Read Holding Register

4 = Read Input Register

6 = Write Single Register

16 = Write Multiple Register

23 = Read/Write Multiple registers

### Physical Layer for MODBUS-RTU:

RS485, 2400 to 115200Baud, 1/8 unit-load, no terminator required, line polarization integrated  
max. 128 devices without repeater

### Physical Layer for MODBUS-TCP:

Ethernet 10BASE-T, RJ45 Socket, Port: 502

Default IP address: 192.168.1.1

### Device Typ:

Modbus RTU Slave, Modbus TCP Server

For additional information about Modbus specifications see <http://www.modbus.org/>

### Default settings:

MODBUS physical values(slave address, baudrate and parity) can set to default settings  
by pressing the reset-button for >5 sec.

After successful reset the status led will flash red for 1 sec.

| Input Register |        |                |               |     |                                                    |
|----------------|--------|----------------|---------------|-----|----------------------------------------------------|
| Signal Name    | Typ    | Modbus address | Default value | Typ | Description                                        |
| dev_ver        | uint16 | 0 (30000)      |               |     | Device version                                     |
| dev_typ        | uint16 | 1              |               |     | Device typ:<br>8 = PLU-K-16.16                     |
| in_value       | uint16 | 2              |               |     | Data of Input<br>.0 Input 1<br>...<br>.15 Input 16 |

|                    |        |    |  |  |                                                                                                  |
|--------------------|--------|----|--|--|--------------------------------------------------------------------------------------------------|
| in_value_hold      | uint16 | 3  |  |  | Data of Input<br>Input set High will be hold until read out<br>.0 Input 1<br>...<br>.15 Input 16 |
| out_man_on         | uint16 | 4  |  |  | Status of Outputs set manual on<br>High = manual on<br>.0 Output 1<br>...<br>.16 Output 16       |
| man_off            | uint16 | 5  |  |  | Status of Outputs set manual off<br>High = manual off<br>.0 Output 1<br>...<br>.15 Output 16     |
| out_value_status   | uint16 | 6  |  |  | Current value of Outputs<br>.0 Output 1<br>...<br>.15 Output 16                                  |
| io_bind_status(0)  | uint16 | 7  |  |  | Stausts of output 1 function:<br>0 = ok<br>1 = output blocked by open/close group                |
| ...                |        |    |  |  |                                                                                                  |
| io_bind_status(15) | uint16 | 22 |  |  | Stausts of output 16 function:<br>0 = ok<br>1 = output blocked by open/close group               |
| in_pulse_count(0)  | uint16 | 23 |  |  | Input 1 pulse (low to high up to 50Hz) count<br>Register resets to zero on read                  |
| ...                |        |    |  |  |                                                                                                  |
| in_pulse_count(15) | uint16 | 38 |  |  | Input 16 pulse (low to high up to 50Hz) count<br>Register resets to zero on read                 |
|                    |        |    |  |  |                                                                                                  |

| Holding Register |        |                |               |      |                                                                                                                                                                                 |
|------------------|--------|----------------|---------------|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Signal Name      | Typ    | Modbus address | Default value | Typ  | Description                                                                                                                                                                     |
| dev_mode         | uint16 | 0 (40000)      | 1             | rem. | Modbus slave mode                                                                                                                                                               |
| dev_address      | uint16 | 1              | 1             | rem. | Modbus slave address                                                                                                                                                            |
| dev_baudrate     | uint16 | 2              | 8             | rem. | Modbus Baudrate:<br>0 = reserved<br>4 = 2400 Baud<br>5 = 4800 Baud<br>6 = 9600 Baud<br>7 = 14400 Baud<br>8 = 19200 Baud<br>9 = 28800 Baud<br>10 = 38400 Baud<br>11 = 57600 Baud |

|                      |        |    |       |      |                                                                                                                                                                                                                                     |
|----------------------|--------|----|-------|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dev_parity           | uint16 | 3  | 2     | rem. | Modbus parity bit:<br>0 = No parity (2bit stop)<br>1 = Odd parity (1bit stop)<br>2 = Even parity (1bit stop)<br>3 = No parity (1bit stop)                                                                                           |
| in_inv               | uint16 | 4  | 0     | rem. | Input inverse:<br>.0 Input 1<br>...<br>.15 Input 16                                                                                                                                                                                 |
| io_bind              | uint16 | 5  | 65535 | rem. | bind in_value to out_value:<br>.0: In/Out 1<br>...<br>.15: In/Out 16                                                                                                                                                                |
| out_value            | uint16 | 6  | 0     |      | Data of Output<br>.0 Output 1<br>...<br>.15 Output 16                                                                                                                                                                               |
| out_inv              | uint16 | 7  | 0     | rem. | Output inverse:<br>.0 Output 1<br>...<br>.15 Output 16                                                                                                                                                                              |
| fault_timer          | uint16 | 8  | 0     | rem. | fault timer<br>0 = disabled<br>>0 = timeout for bus fault (1 step = 50ms)<br>On bus fault, out_value will be set to<br>fault_out_value                                                                                              |
| fault_out_value      | uint16 | 9  | 0     | rem. | Data of Output on fault<br>.0 Output 1<br>...<br>.15 Output 16                                                                                                                                                                      |
| io_bind_function(0)  | uint16 | 10 | 0     | rem. | function for output 1 if io_bind is set:<br>0 = output set directly by input<br>1 = output will toggle by input low to high<br>2 = output set directly by input,<br>but even and odd outputs are locked<br>(open/close group)       |
| ...                  |        |    |       |      |                                                                                                                                                                                                                                     |
| io_bind_function(15) | uint16 | 25 | 0     | rem. | function for output 16 if io_bind is set:<br>0 = output set directly by input<br>1 = output will toggle by input low to high<br>2 = output set directly by input,<br>but even and odd outputs are locked<br>(open/close group)      |
| io_bind_control(0)   | uint16 | 26 | 65535 |      | Output 1 remote control if io_bind is set:<br>0 = set output off once<br>1 = set output on once<br>2 = toggle output once<br>3 = force output off<br>4 = force output on<br>65534 = function not available<br>65535 = function done |
| ...                  |        |    |       |      |                                                                                                                                                                                                                                     |

|                     |        |    |       |  |                                                                                                                                                                                                                                      |
|---------------------|--------|----|-------|--|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| io_bind_control(15) | uint16 | 41 | 65535 |  | Output 16 remote control if io_bind is set:<br>0 = set output off once<br>1 = set output on once<br>2 = toggle output once<br>3 = force output off<br>4 = force output on<br>65534 = function not available<br>65535 = function done |
|                     |        |    |       |  |                                                                                                                                                                                                                                      |
|                     |        |    |       |  |                                                                                                                                                                                                                                      |

rem. = remanence data