

Version 2.1 Update notes

- Modbus-TCP Server:

The FP Web-Server can be accessed by a Modbus-TCP client (SCADA) via Ethernet for PLC data communication. The FP Web-Server requests PLC data (MEWTOCOL protocol) via 3-pin or/and 9-pin RS232C port. The 9-pin RS232C can alternatively communicate via Modbus-RTU protocol with a Modbus-RTU slave unit (gateway function routes all Modbus requests).



An integrated setup wizard helps to configure the Modbus-TCP server. A Modbus to FP-Series PLC address translation table can be edited. For details on the implemented Modbus functions and PLC offsets, see below.

- Modbus-TCP Client:

Either the PLC or a Modbus-RTU master unit can control the Modbus-TCP client functions of the FP Web-Server.

a.) For PLC control a shared buffer in the PLC's memory is used. The PLC places the server IP address and the Modbus commands in this buffer. The FP Web-Server reads the buffer and executes the commands.

Additionally the PLC can also access a Modbus-RTU slave unit via 9-pin RS232C.

b.) Any Modbus-RTU master connected to the 9-pin RS232C of the FP Web-Server can access a remote Modbus-TCP server. This gateway functions routes all Modbus-RTU function requests to the Modbus-TCP server. Optionally the Modbus-RTU master can address different Modbus-TCP servers (IP address and/or unit ID is calculated by the accessed RTU slave address) or the local PLC directly. An integrated setup wizard helps to configure the Modbus-TCP client. For details on the implemented Modbus functions and PLC offsets, see below.

- The FP Web-Server can dial up the Internet (via GPRS) and can register its new Internet IP address at a central Internet server. Three modes are possible:

- Log onto a standard Http server. A programmable PHP script generates a Html list.

For more information please, contact Panasonic Electric Works.

- Log onto and update the DynDNS server (see <http://www.dyndns.org/>)

- Log onto an Emazy server (see <http://www.emazy.com/>)

- Updated IEC60870 library. New function block M_IEC_BO_TB_LOG for Class1 or Class2 redirection. With this it is possible to compact binary information.

- Additionally we updated/included the following minor functions:

- The modem dial-in gateway (PPP server) can now also handle a second user account (user name and password).

- PLC sample programs for Modbus-TCP and GPRS dial-up included.

- PLC real-time clock update via NTP server for FP2 optimized

- MEWTOCOL port server memory size optimized

- PLC MEWTOCOL station address for IEC60870 and other minor IEC60870 changes

- Summary list of enabled main functions in the "Config" tab

- Start of the Jar archive editor optimized for JarEdit v1.1 and JRE v5

- Some parameter entry dialogs optimized (Email server, UDP port, PPP-Server ...)

Following Modbus-TCP functions and PLC offsets are implemented:

- 02 (0x02) Read Discrete Inputs
 - Read "X" inputs of the PLC
 - Bit address starts at: 100001 = 0x0000 = X0
 - Maximum of 2000 inputs per request
- 01 (0x01) Read Coils (Custom functions 66, 68)
 - Read "R" relays of the PLC
 - Read "Y" outputs of the PLC if start address \geq Offset_1 (15000)
 - Read "L" relays of the PLC if start address \geq Offset_2 (25000)
 - Bit address starts at: 000001 = 0x0000 = R0
 - Maximum of 2000 coils per request
- 05 (0x05) Write Single Coil
 - Write one "R" relay in the PLC
 - Write one "Y" output in the PLC if start address \geq Offset_1 (15000)
 - Write one "L" relay in the PLC if start address \geq Offset_2 (25000)
 - Bit address starts at: 000001 = 0x0000 = R0
- 15 (0x0F) Write Multiple (Custom functions 67, 69)
 - Write "R" relays in der SPS
 - Write "Y" outputs in the PLC if start address \geq Offset_1 (15000)
 - Write "L" relays in the PLC if start address \geq Offset_2 (25000)
 - Bit address starts at: 000001 = 0x0000 = R0
 - Maximum of 1968 coils per request
- 04 (0x04) Read Input Registers
 - Read "WX" input register of the PLC
 - Word address starts at: 300001 = 0x0000 = WX0
 - Maximum of 125 registers per request
- 03 (0x03) Read Holding Registers (Custom functions 100,102,104,106,108)
 - Read "DT" Data register of the PLC
 - Read "FL" register of the PLC if start address \geq Offset_3 (12000)
 - Read "LD" register of the PLC if start address \geq Offset_4 (45000)
 - Read "WR" register of the PLC if start address \geq Offset_5 (55000)
 - Read "WY" register of the PLC if start address \geq Offset_6 (56000)
 - Read "WL" register of the PLC if start address \geq Offset_7 (57000)
 - Word address starts at: 400001 = 0x0000 = DT0
 - Maximum of 125 registers per request
- 06 (0x06) Write Single Register
 - Write one "DT" data register in the PLC
 - Write one "FL" register in the PLC if start address \geq Offset_3 (12000)
 - Write one "LD" register in the PLC if start address \geq Offset_4 (45000)
 - Write one "WR" register in the PLC if start address \geq Offset_5 (55000)
 - Write one "WY" register in the PLC if start address \geq Offset_6 (56000)
 - Write one "WL" register in the PLC if start address \geq Offset_7 (57000)
 - Word address starts at: 400001 = 0x0000 = DT0
- 16 (0x10) Write Multiple Registers (Custom functions 101,103,105,107,109)
 - Write "DT" data registers in the PLC
 - Write "FL" registers in the PLC if start address \geq Offset_3 (12000)
 - Write "LD" registers in the PLC if start address \geq Offset_4 (45000)
 - Write "WR" registers in the PLC if start address \geq Offset_5 (55000)
 - Write "WY" registers in the PLC if start address \geq Offset_6 (56000)
 - Write "WL" registers in the PLC if start address \geq Offset_7 (57000)
 - Word address starts at: 400001 = 0x0000 = DT0
 - Maximum of 123 registers per request
- 43/14 (0x2B/0x0D) Read Device Identification
 - Read unit identification text (fixed in firmware)