lamaPLC: RP2040_ETH_Modul: Modbus TCP example

The RP2040 module has Ethernet integrated, which can be accessed via UART1. This code creates Modbus Holding registers on the module with contents from 1 to 10.

Important: The Ethernet module is accessible by **RP2040_ETH** via **UART1** with the following configuration:

uart1 = UART(1, baudrate=115200, tx=Pin(20), rx=Pin(21), timeout=50)

```
from machine import UART, Pin
import time

# --- Konfiguráció ---
uart1 = UART(1, baudrate=115200, tx=Pin(20), rx=Pin(21), timeout=50)

# Regiszterek inicializálása: 10 darab regiszter (index 0..9)
# Értékek: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
holding_registers = [i + 1 for i in range(10)]

print(f"Modbus TCP Szerver indul... Regiszterek: {holding_registers}")

while True:
    if uart1.any():
        raw_data = uart1.read()

        # Modbus TCP minimális hossza: 12 bájt
        if len(raw_data) >= 12:
            header = raw_data[:6]   # MBAP Header
            pdu = raw_data[6:]      # UnitID + PDU

            unit_id = pdu[0]
            func_code = pdu[1]

            # Start Address (pdu[2:4]) és Quantity (pdu[4:6])
            start_addr = (pdu[2] << 8) | pdu[3]
            quantity = (pdu[4] << 8) | pdu[5]

            # 03: Read Holding Registers
            if func_code == 3:
                # Ellenőrizzük, hogy a kért tartomány érvényes-e (0-9
között)
                if start_addr + quantity <= len(holding_registers):
```

```
                    # Adatbájtok összeállítása
                    byte_count = quantity * 2
                    data_payload = bytearray()

                    for i in range(start_addr, start_addr + quantity):
                        val = holding_registers[i]
                        data_payload.append((val >> 8) & 0xFF) # High byte
                        data_payload.append(val & 0xFF)        # Low byte

                    # PDU válasz: UnitID, Func, ByteCount, Adatok
                    pdu_res = bytearray([unit_id, func_code, byte_count]) +
data_payload

                    # TCP Header frissítése (hossz: UnitID(1) + PDU többi
része)
                    res_header = bytearray(header)
                    res_header[4] = 0x00
                    res_header[5] = len(pdu_res)

                    uart1.write(res_header + pdu_res)
                    print(f"Kérés: Addr {start_addr}, Qty {quantity} ->
Válasz elküldve.")
                else:
                    # Exception 02: Illegal Data Address (ha túl sokat kérne
a Master)
                    pdu_res = bytearray([unit_id, func_code + 0x80, 0x02])
                    res_header = bytearray(header)
                    res_header[5] = 3
                    uart1.write(res_header + pdu_res)
                    print("Hiba: Érvénytelen regiszter cím!")

    time.sleep(0.01)
```

code!, micropython, 2026, RP2040 ETH, Modbus, test