# LISTADO DE MATERIALES (BOM) — MÓDULO 05

## Hardware de Campo, Pasarela Edge Raspberry Pi, Red RS-485 y Software Broker MQTT

### 1. Unidad Central de Borde (Gateway IIoT & Broker MQTT)

| Ítem | Cant. | Componente / Especificación Técnica | Función en la Arquitectura del Módulo 05 |
|---|---|---|---|
| **01** | 1 | **Raspberry Pi 4 Model B (4 GB / 8 GB RAM)** o **Raspberry Pi 5** | Ejecuta en simultáneo el Broker **Eclipse Mosquitto 2.x** y el demonio **Puente Python Modbus ↔ MQTT**. |
| **02** | 1 | **Fuente Oficial USB-C 5.1 V / 3.0 A (15.3 W)** o Riel DIN 5 V / 4 A | Alimentación regulada libre de caídas de tensión durante ráfagas de red y escritura en disco. |
| **03** | 1 | **Tarjeta microSD Industrial A2 / U3 (32 GB)** o SSD NVMe por USB 3.0 | Almacenamiento de Raspberry Pi OS (64-bit), base de persistencia `mosquitto.db` y registros `systemd`. |
| **04** | 1 | **Carcasa de Aluminio Pasiva o Riel DIN** + Cable Ethernet Cat 6 RJ45 | Disipación térmica continua 24/7 y enlace cableado determinista hacia el switch de planta. |

### 2. Interfaz de Acoplamiento Físico RS-485 (Borde ↔ Campo)

| Ítem | Cant. | Componente / Especificación Técnica | Función en la Arquitectura del Módulo 05 |
|---|---|---|---|
| **05** | 1 | **Conversor Industrial USB a RS-485** (Chipset FT232RL o CH340G con TVS) | Expone el puerto `/dev/ttyUSB0` en Linux con bornera de tornillo ( `A+` , `B-` , `GND` ). |
| **06** | 1 | **Módulo Transceptor MAX485CPA** (DIP-8 en protoboard o PCB con bornera) | Convierte niveles UART TTL ($0–5\ V$) del Arduino UNO a señal diferencial RS-485 ($\pm 1.5\ V$ a $\pm 5\ V$). |
| **07** | 2 | **Resistencias de Terminación** $R_T = 120\ \Omega$ **(**$1/4\ W$**, 1%)** | Adaptación de impedancia característica ($Z_0 = 120\ \Omega$) en ambos extremos físicos del bus. |
| **08** | 2 | **Resistencias de Polarización Fail-Safe (**$680\ \Omega$**,** $1/4\ W$**)** | Mantienen $V_{AB} > +200\ mV$ cuando todos los nodos están en alta impedancia ( `DE=0` ). |
| **09** | 5 m | **Cable de Par Trenzado Apantallado (STP Belden 9841 / Cat 5e)** | Transporte diferencial de las líneas `A+` , `B-` y drenaje de referencia `GND` . |

### 3. Nodo Esclavo de Campo en Protoboard (Arduino UNO ID=1 + Sensores y Actuadores)

| Ítem | Cant. | Componente / Especificación Técnica | Pin Arduino / Registro Modbus / Tópico MQTT |
|---|---|---|---|
| **10** | 1 | **Arduino UNO R3 (ATmega328P, 16 MHz)** con cable USB-B | Esclavo Modbus RTU ( `ID=1` ): pines `D0 (RX)` , `D1 (TX)` y `D2 (DE/RE)` hacia el MAX485. |
| **11** | 2 | **Protoboards (Solderless Breadboards)** de 830 y 400 puntos | Montaje físico de relés, transceptor, divisores resistivos, pulsadores y diodos LED testigo. |
| **12** | 2 | **Relés de 5 V DC ( `K52B-M-DC5` / Módulo 2 Canales Optoacoplado)** + `2N2222` + `1N4007` | Conmutación de potencia controlada por `D8` (**Reg 40003**, Extractor) y `D4` (**Reg 40004**, Bomba). |
| **13** | 2 | **Motores DC de 12 V con Engranaje Reductor / Ventilador Brushless** | Actuadores electromecánicos de extracción forzada y bombeo gobernados desde MQTT ( `0x06` ). |
| **14** | 1 | **Sensor de Temperatura NTC** $10\ k\Omega$ **/ Sonda Impermeable DS18B20** | Entrada `A0` : temperatura en décimas de grado (**Reg 40001** → `.../sensores/temperatura` ). |
| **15** | 1 | **Fotorresistor LDR GL5528** + Resistencia $10\ k\Omega$ (Divisor) | Entrada `A1` : porcentaje de radiación lumínica (**Reg 40002** → `.../sensores/iluminacion` ). |
| **16** | 1 | **Sensor Ultrasónico HC-SR04** (Nivel de estanque de riego) | Pines `D3 (TRIG)` y `A2 (ECHO)` : nivel de agua en cm (**Reg 40005** → `.../sensores/nivel_tanque` ). |
| **17** | 1 | **Micro Servomotor SG90 (PWM** $50\ Hz$**)** | Pin `D5` : apertura proporcional de escotilla cenital (**Reg 40006**). |
| **18** | 4 | **Diodos LED 5 mm** (Azul, Verde, Ámbar, Rojo) + 4 Resistencias $220\ \Omega$ | Pines `D9..D12` : indicadores visuales de tráfico MQTT, estado Online, trama Modbus y Alarma. |
| **19** | 2 | **Pulsadores Táctiles** $6\times 6\ mm$ + 2 Resistencias Pull-Down $10\ k\Omega$ | Pines `D6` y `D7` : parada local de emergencia y rearme manual en protoboard. |
| **20** | 1 | **Fuente de Banco / Auxiliar DC (**$12\ V$ **/** $5\ V$**)** con masa común ( `GND` ) | Alimentación dedicada para bobinas de relés, servomotor y motores DC sin inducir caída en el MCU. |

## 4. Pila de Software e Instrumentación Lógica

| Paquete / Herramienta | Versión Recomendada | Propósito en el Laboratorio |
|---|---|---|
| **Eclipse Mosquitto** | `>= 2.0.18` | Broker MQTT ligero en C con soporte TLS 1.3, autenticación `SHA-512` y listas ACL. |
| **mosquitto-clients** | `>= 2.0.18` | Herramientas CLI `mosquitto_pub` , `mosquitto_sub` y `mosquitto_passwd` . |
| **Python 3 + MinimalModbus** | `Python 3.11+` / `2.1.1` | Maestro Modbus RTU sobre `/dev/ttyUSB0` con cálculo y verificación automática de CRC-16. |
| **Eclipse Paho-MQTT Python** | `>= 1.6.1 / 2.0` | Cliente asíncrono MQTT con soporte nativo de `will_set` (LWT), reconexión y niveles QoS 0/1/2. |