Manual de instalación y programación · JWPLC-BASIC-R2

JWPLC Basic

Instalación, cableado y programación · JW Control ESP32 Boards v2.1.0-alpha.14 · hardware v2.0.0

> Este manual cubre la instalación mecánica, el cableado de entradas y salidas, y la programación en Arduino IDE con el package v2.1.0-alpha.14. Los datos eléctricos detallados están en la ficha técnica del JWPLC Basic. Aplica a los formatos JWPLC Basic, JWPLC Core y JWPLC I/O; las diferencias se indican donde corresponde.

Índice

1. Instalación mecánica 2
1.1 Montaje en riel DIN 2
1.2 Dimensiones de montaje 2
2. Asignación de bornes 2
2.1 Alimentación 2
2.2 Entradas digitales (I0.0–I0.7) 2
2.3 Salidas a relé (Q0.0–Q0.7) 2
2.4 Puerto RS-485 3
3. Cableado eléctrico 3
3.1 Fuente de alimentación 3
3.2 Cableado de entradas digitales 3
3.3 Cableado de salidas a relé 4
3.4 RS-485 y Modbus RTU 4
3.5 Guía rápida de cargas para los relés 4
4. Programación con Arduino IDE 4
4.1 Instalación del package 5
4.2 Placas 5
4.3 Entradas y salidas 5
4.4 Ejemplos de código 5
4.5 Ejemplos incluidos en el package 6
4.6 Programación en escalera con OpenPLC 7
4.7 Diseño de pantallas con JWPLC HMI Designer 7
5. Periféricos integrados: referencia rápida 7
6. Registro de revisiones 7

1. Instalación mecánica

1.1 Montaje en riel DIN

1. Verifica que el riel DIN de 35 mm sea estándar (IEC 60715). El equipo necesita al menos 110 mm libres en el riel.
2. Engancha la parte superior del equipo al riel y presiona hacia abajo hasta que el clip inferior encaje con un clic.
3. Deja 50 mm libres por encima y por debajo, para ventilación y cableado.
4. Instálalo en posición horizontal, con la etiqueta frontal visible.

1.2 Dimensiones de montaje

Dimensión	Valor
Ancho	110 mm (127 mm con la antena Wi-Fi exterior)
Alto	132 mm
Profundidad	70 mm, con el riel DIN
Peso	≈ 400 g
Riel DIN	35 mm estándar
Espacio libre mínimo	50 mm arriba y abajo
Orientación	Horizontal

2. Asignación de bornes

> Las borneras son enchufables y sus etiquetas están serigrafiadas en la tarjeta. Sección admitida en todos los bloques: 0.2–2.5 mm² (28–12 AWG).

2.1 Alimentación

Pin	Etiqueta	Descripción
1	L+	Alimentación 24 V DC positivo
2	M	Alimentación 0 V DC (masa común)

2.2 Entradas digitales (I0.0–I0.7)

Pin	Etiqueta	Descripción
1	1M	Común de entradas: conectar a M / 0 V de los sensores
2	I0.0	Entrada digital 0, 24 V DC, sinking; para sensores PNP/sourcing
3	I0.1	Entrada digital 1
4	I0.2	Entrada digital 2
5	I0.3	Entrada digital 3
6	I0.4	Entrada digital 4
7	I0.5	Entrada digital 5
8	I0.6	Entrada digital 6
9	I0.7	Entrada digital 7

2.3 Salidas a relé (Q0.0–Q0.7)

Pin	Etiqueta	Descripción
1	Q0.0 COM	Común del relé 0 (contacto seco)
2	Q0.0 NO	Normalmente abierto del relé 0
3	Q0.1 COM	Común del relé 1 (contacto seco)
4	Q0.1 NO	Normalmente abierto del relé 1
5	Q0.2 COM	Común del relé 2 (contacto seco)
6	Q0.2 NO	Normalmente abierto del relé 2

Pin	Etiqueta	Descripción
7	Q0.3 COM	Común del relé 3 (contacto seco)
8	Q0.3 NO	Normalmente abierto del relé 3
9	Q0.4 COM	Común del relé 4 (contacto seco)
10	Q0.4 NO	Normalmente abierto del relé 4
11	Q0.5 COM	Común del relé 5 (contacto seco)
12	Q0.5 NO	Normalmente abierto del relé 5
13	Q0.6 COM	Común del relé 6 (contacto seco)
14	Q0.6 NO	Normalmente abierto del relé 6
15	Q0.7 COM	Común del relé 7 (contacto seco)
16	Q0.7 NO	Normalmente abierto del relé 7

2.4 Puerto RS-485

Pin	Etiqueta	Descripción
1	GND	Masa de referencia RS-485
2	A (D+)	Línea diferencial positiva
3	B (D−)	Línea diferencial negativa
4	SHIELD	Blindaje del cable

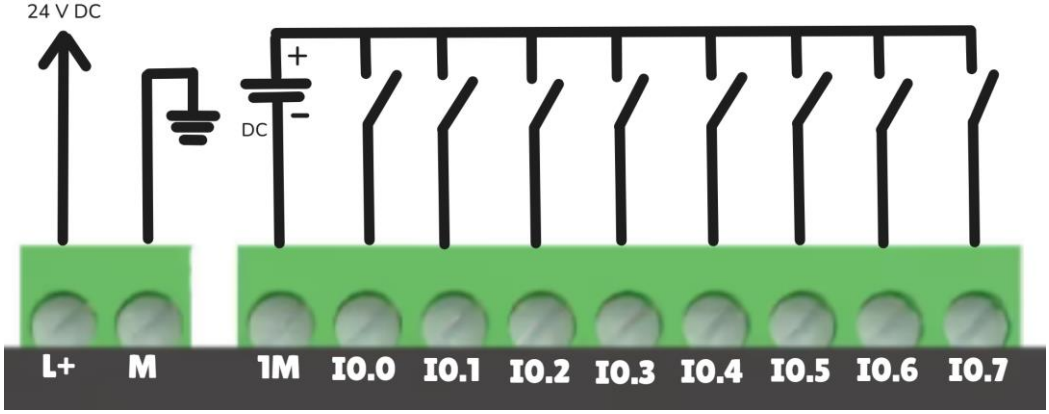
3. Cableado eléctrico

3.1 Fuente de alimentación

- Usa una fuente regulada de 24 V DC con al menos 20 % de margen sobre la corriente máxima calculada (mínimo 250 mA a plena carga).
- Con cargas de alta corriente (contactores, motores), usa una fuente para la lógica y otra para las cargas de los relés.
- El equipo tiene diodo en serie contra polaridad inversa y varistor contra sobretensiones, pero no reemplaza un fusible externo.
- **Fusible recomendado:** 1–2 A de acción rápida (tipo F) en la línea L+.

3.2 Cableado de entradas digitales

- Las entradas son de tipo sinking. El común 1M se conecta a M / 0 V.
- Un sensor PNP/sourcing entrega +24 V a I0.x cuando está activo; la entrada se lee como HIGH.
- Un sensor NPN/sinking no se conecta directo: necesita una interfaz, un relé o un convertidor.
- Contacto seco: L+ → contacto → I0.x, y 1M → M.
- Longitud de cable recomendada: hasta 100 m con cable apantallado de 24 AWG.

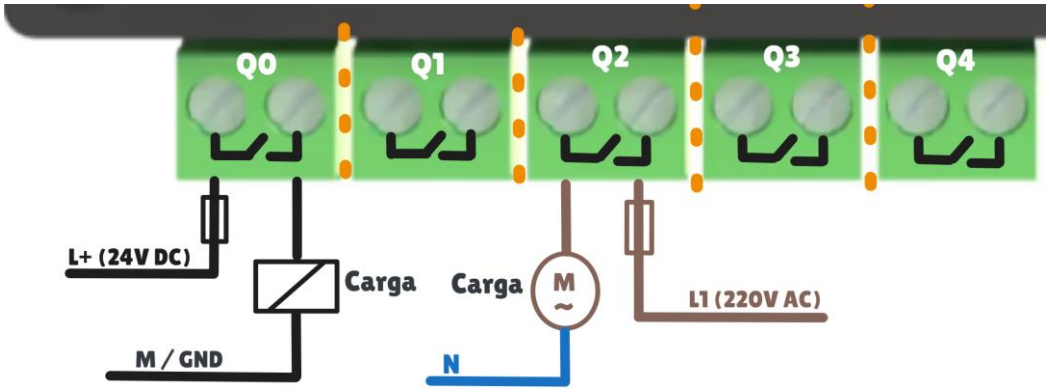


Cableado típico de entradas sinking con contactos secos y fuente de 24 V DC.

3.3 Cableado de salidas a relé

- Cada salida es un contacto seco independiente (COM y NO); la fuente de la carga es externa al equipo.
- No superes 5 A por canal ni 10 A en total (suma de relés activos).
- Pon un fusible por canal en la línea de la carga.
- El RS-485 y las entradas van en canaletas separadas del cableado de fuerza de los relés.

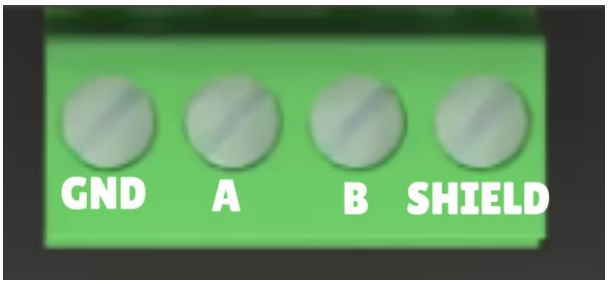
Cargas inductivas: instala supresión en paralelo con la carga (diodo 1N4007 en DC, varistor MOV275 en AC). Sin ella, los picos de tensión acortan la vida del contacto y pueden interferir con el propio controlador.



Cargas de 24 V DC (Q0) y 220 V AC (Q2) con sus fusibles externos.

3.4 RS-485 y Modbus RTU

- Usa par trenzado apantallado (24 AWG) con impedancia característica de 120 Ω.
- La terminación de 120 Ω está integrada y es configurable: actívala solo en los dos extremos del bus.
- Conecta el blindaje en un solo extremo, a la tierra del tablero o del controlador.
- En tramos largos (más de 300 m), usa 9600 bps.



Bus RS-485 en cadena, con el par trenzado y el blindaje.

3.5 Guía rápida de cargas para los relés

Tipo de carga	I máx.	Frec. máx.	Vida útil	Protección recomendada
Sin carga (maniobra)	—	3 Hz	20 × 10⁶	Sin protección adicional
Resistiva (DC/AC)	5 A	0.33 Hz	100 000	Fusible externo
Inductiva (AC-15)	2 A	0.1 Hz	100 000	Diodo o varistor obligatorio
Lámpara incandescente	1 A	0.1 Hz	50 000	Fusible lento (pico de arranque 10×)
Lámpara LED	2 A	1 Hz	150 000	Verificar la corriente de arranque del driver
Bobina de contactor	1.5 A	0.1 Hz	80 000	Diodo en paralelo
Motor pequeño (DC)	1 A	0.05 Hz	30 000	Diodo y varistor

4. Programación con Arduino IDE

Al elegir la placa, el equipo arranca con sus periféricos listos: no hace falta configurar pines, iniciar la pantalla ni levantar la red desde el programa. Tu programa se ocupa solo de la lógica de la máquina.

4.1 Instalación del package

Recomendado: Arduino IDE 2.3.10 o superior, que recompila un proyecto en la mitad del tiempo que las versiones anteriores.

1. En Arduino IDE abre Archivo → Preferencias.
2. En «URLs adicionales del gestor de tarjetas» pega uno de estos índices:

   Estable (v2.0.0), para proyectos en producción:

   ```
   https://raw.githubusercontent.com/JW-Control/platform-jwplc/main/JWPLC/package_jwplc_index.json
   ```

   Desarrollo (v2.1.0-alpha.14), para probar las novedades de la línea 2.1:

   ```
   https://raw.githubusercontent.com/JW-Control/platform-jwplc/main/JWPLC/package_jwplc_index_dev.json
   ```

3. Abre Herramientas → Placa → Gestor de tarjetas, busca «JW Control ESP32 Boards» e instala la versión que quieras.
4. Elige la placa en Herramientas → Placa → JW Control ESP32 Boards, y el puerto que aparece al conectar el cable USB-C.

Con Arduino CLI:

```
arduino-cli config add board_manager.additional_urls https://raw.githubusercontent.com/JW-
Control/platform-jwplc/main/JWPLC/package_jwplc_index_dev.json
arduino-cli core update-index
arduino-cli core install jwplc:esp32@2.1.0-alpha.14
```

> No instales a mano librerías con los mismos nombres que trae el package: el package ya incluye las versiones probadas con el JWPLC.

4.2 Placas

Placa	FQBN	Qué incluye
JWPLC Basic	jwplc:esp32:jwplcbasic	El equipo completo, con Ethernet, FRAM y microSD
JWPLC Basic Core	jwplc:esp32:jwplcbasiccore	Para el JWPLC Core y el JWPLC I/O: sin Ethernet, FRAM ni microSD; con reloj
ESP32 Board	jwplc:esp32:esp32	ESP32 genérico, sin periféricos JWPLC

4.3 Entradas y salidas

> No hace falta pinMode() para las entradas (I0_x) ni para las salidas (Q0_x): el package inicializa los expansores al arrancar.

Función	Descripción
I0_0 … I0_7	Entradas digitales 0–7
Q0_0 … Q0_7	Salidas a relé 0–7
digitalRead(I0_x)	Lee una entrada (HIGH o LOW)
digitalWrite(Q0_x, estado)	Activa o desactiva un relé
JWPLC_readInputs()	Lee las 8 entradas como un byte (bit 0 = I0_0)
JWPLC_writeOutputs(mascara)	Escribe las 8 salidas en un solo paso
JWPLC_IO	Estado de entradas y salidas leído por el package, sin una lectura extra

4.4 Ejemplos de código

Ejemplo 1. La salida Q0.0 copia la entrada I0.0:

```
void setup()
{
}

void loop()
{
    digitalWrite(Q0_0, digitalRead(I0_0));
}
```

Ejemplo 2. La tecla OK enciende y apaga Q0.0, con un pitido:

```
bool encendida = false;

void setup()
{
}

void loop()
{
    if (JWPLC_Buttons.pressed(BTN_OK))
    {
        encendida = !encendida;
        digitalWrite(Q0_0, encendida ? HIGH : LOW);
        JWPLC_Buzzer.beep();
    }
}
```

Ejemplo 3. Ethernet con DHCP (solo JWPLC Basic):

```
void setup()
{
    Serial.begin(115200);
    JWPLC_Ethernet.useDHCP();
}

void loop()
{
    if (JWPLC_Ethernet.isReady())
    {
        Serial.println(JWPLC_Ethernet.localIP());
        delay(5000);
    }
}
```

Ejemplo 4. Servidor Modbus TCP (solo JWPLC Basic):

```
#include <JWPLC_ModbusTCP.h>

uint16_t registros[16] = {};

void setup()
{
    JWPLC_ModbusTCP.setHoldingRegisters(registros, 16);
    JWPLC_ModbusTCP.beginServer(1, 502);
}

void loop()
{
    registros[0] = digitalRead(I0_0);
}
```

Ejemplo 5. Esclavo Modbus RTU por RS-485:

```
uint16_t registros[8] = {};

void setup()
{
    JWPLC_ModbusRTU.setHoldingRegisters(registros, 8);
    JWPLC_ModbusRTU.begin(2, 115200, SERIAL_8N1);
}

void loop()
{
    JWPLC_ModbusRTU.task();
}
```

4.5 Ejemplos incluidos en el package

Están en Archivo → Ejemplos. Los 78 ejemplos del package compilan en las placas JWPLC Basic y JWPLC Basic Core.

Librería	Ejemplos principales
JWPLC_GlobalPeripherals	Entradas y salidas, botonera, reloj, FRAM, microSD y buzzer

Librería	Ejemplos principales
JWPLC_Display	Pantalla en reposo, campos y páginas HMI, dibujo directo y plantillas: caldera, botonera interactiva y planta con iconos
JWPLC_Ethernet	DHCP, IP fija y diagnóstico
JWPLC_ModbusTCP	Servidor y cliente
JWPLC_ModbusRTU	Esclavo con registros, maestro que lee y maestro que escribe
JWPLC_RS485	Comunicación serie por RS-485

4.6 Programación en escalera con OpenPLC

- OpenPLC Editor JWPLC Edition es una integración opcional que se instala aparte, con su propio instalador.
- Permite programar en los lenguajes IEC 61131-3: LD, FBD, IL, ST y SFC.
- El mapa de entradas y salidas del JWPLC Basic es compatible con el runtime de OpenPLC.

4.7 Diseño de pantallas con JWPLC HMI Designer

Aplicación gratuita para Windows que diseña las pantallas del JWPLC de forma visual: textos, valores, indicadores, barras, menús, figuras e imágenes, en una o varias páginas. Escribe en la carpeta del programa el archivo JWPLC_HMI_Generated.h, se abre desde Arduino IDE y muestra una vista previa en vivo conectada al equipo. No aplica al JWPLC I/O, que no tiene pantalla.

5. Periféricos integrados: referencia rápida

Todos se usan con objetos globales ya creados, sin #include. La única excepción es Modbus TCP, que se agrega con #include <JWPLC_ModbusTCP.h>. No crees una segunda instancia del mismo hardware.

Periférico	Objeto global	Notas
Pantalla TFT **	JWPLC_Display, JWPLC_TFT	ST7789 de 320 × 170; reposo, páginas HMI, menús y dibujo directo
Botonera **	JWPLC_Buttons	pressed(), released() e isDown() con BTN_LEFT, BTN_UP, BTN_RIGHT, BTN_DOWN, BTN_OK y BTN_ESC
Buzzer **	JWPLC_Buzzer	beep(), tone() y play() sin detener el programa; volumen de 0 a 255
Reloj (RTC)	JWPLC_Time, JWPLC_RTC	DS3232M; JWPLC_Time da fecha y hora sin una lectura I²C extra
FRAM *	JWPLC_FRAM	8 KB no volátil; lectura y escritura por byte o por bloque
microSD *	JWPLC_SD	FAT32, compatible con la librería SD estándar
Ethernet *	JWPLC_Ethernet	W5500; DHCP o IP fija, TCP y UDP
RS-485	JWPLC_RS485	MAX13487, half-duplex con dirección automática
Modbus RTU	JWPLC_ModbusRTU	Maestro o esclavo; FC01–FC06, FC15 y FC16
Modbus TCP *	JWPLC_ModbusTCP	Servidor o cliente sobre Ethernet

* Solo en el formato JWPLC Basic. ** No incluido en el formato JWPLC I/O.

Soporte técnico y documentación: jw-control.com · jw.control.peru@gmail.com · WhatsApp +51 918 449 114 · github.com/JW-Control/platform-jwplc

6. Registro de revisiones

Revisión	Fecha	Descripción
1.0	2026-08	Borrador inicial, separado del documento maestro.
2.0	2026-10	Formatos JWPLC Basic, JWPLC Core y JWPLC I/O. Package v2.1.0-alpha.14: índices estable y de desarrollo, Arduino IDE 2.3.10, placa Basic Core, botonera, buzzer, Ethernet, Modbus TCP, ejemplos incluidos y JWPLC HMI Designer.