

1. Descripción general.

ArduPLCm es un dispositivo microcontrolador que nos ofrece la posibilidad de conectar y controlar dispositivos eléctricos o electrónicos directamente sin necesidad de componentes adicionales. Básicamente es un Arduino con posibilidad real de control.

Gracias a su puerto RS485, está recomendado para aplicaciones distribuidas como domótica, conexión con PLCs industriales, control, etc.

Todas las entradas y salidas son accesibles a través de robustas bornas de tornillo.

ArduPLCm se entrega montado en una caja estándar para carril DIN. Esto nos facilita el montaje dentro de cuadros eléctricos para automatización, domótica, etc.

2. Características.

- Microcontrolador Atmel ATmega328.
- 4 relés de salida.
- 4 entradas digitales.
- 4 entradas analógicas.
- Dipswitch 6 posiciones.
- Puerto de comunicaciones RS485.
- Programación por ICSP o conector serie TTL.
- Amplio rango de alimentación de 6.5 a 30 voltios.
- Regulador conmutado.
- Compatible con Arduino Duemilanove.
- Caja para carril DIN.

3. Microcontrolador

ArduPLCm está basado en Arduino y por consiguiente podremos utilizar cualquiera de los entornos de desarrollo Arduino para programar nuestro dispositivo.

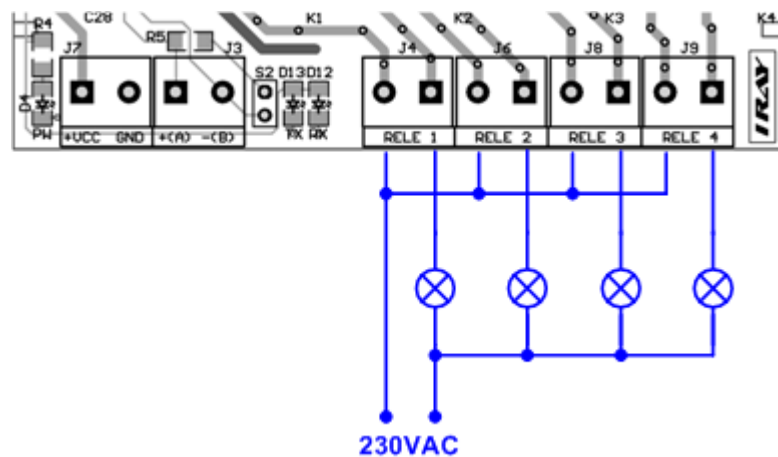
La equivalencia entre las E/S de ArduPLCm y Arduino son las siguientes:

E/S ArduPLCm	E/S Arduino
RELE 1	10
RELE 2	11
RELE 3	12
RELE 4	13
ENTRADA DIGITAL DIN1	2
ENTRADA DIGITAL DIN2	3
ENTRADA DIGITAL DIN3	18
ENTRADA DIGITAL DIN4	19
ENTRADA ANALÓGICA AIN1	CANAL 0
ENTRADA ANALÓGICA AIN2	CANAL 1
ENTRADA ANALÓGICA AIN3	CANAL 2
ENTRADA ANALÓGICA AIN4	CANAL 3
DIPSWITCH 1	4
DIPSWITCH 2	5
DIPSWITCH 3	6
DIPSWITCH 4	7
DIPSWITCH 5	8
DIPSWITCH 6	9

4. Relés de salida.

Los relés de salida son del tipo normalmente abierto y con sus comunes totalmente independientes.

Ejemplo de conexión para controlar 4 lámparas:



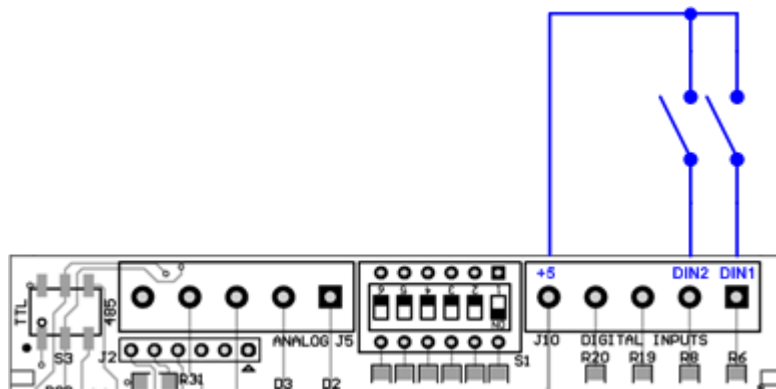
Características técnicas de las salidas de relé:

- Máxima corriente: 10A
- Máxima tensión: 250VAC o 30VDC
- Máxima potencia: 2500VA, 300W
- Resistencia del contacto: 100mOhm
- Vida eléctrica: 100,000 operaciones a 10A y 250VAC
- Vida mecánica: 10,000,000 operaciones

5. Entradas digitales.

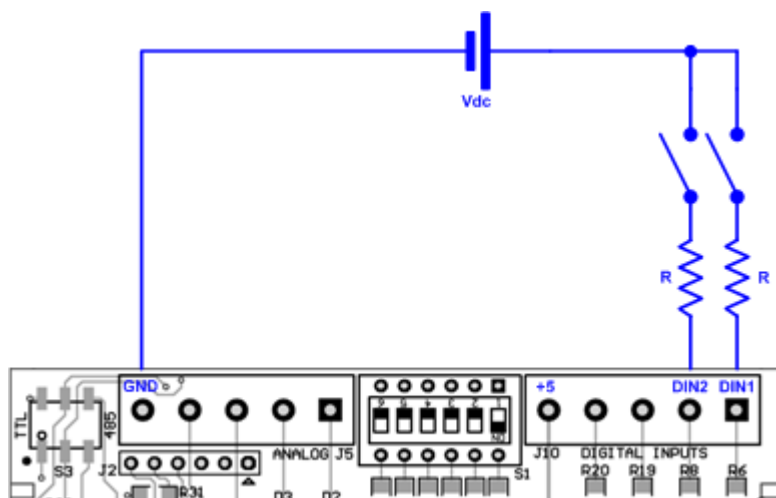
Las entradas digitales se activan con un nivel lógico de 5VDC referenciado a GND.

Un ejemplo de conexión para 2 interruptores sería de la siguiente manera:



Puede utilizar otros niveles de tensión como 12 o 24V pero para ello debe intercalar una resistencia en serie con la entrada de la siguiente forma:

NIVEL LÓGICO	RESISTENCIA
5VDC	no necesario
12VDC	3K3
24VDC	10K

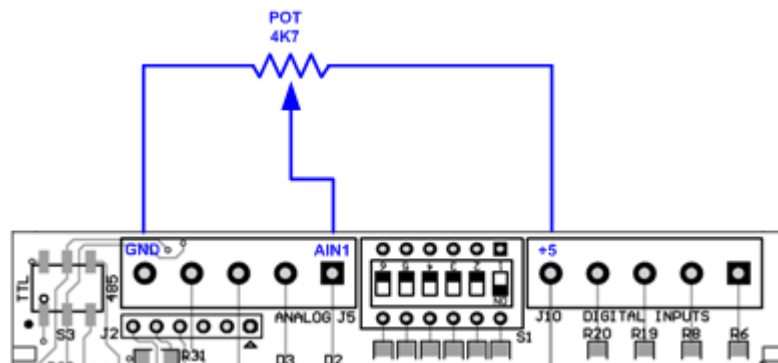


6. Entradas analógicas.

Disponemos de 4 canales analógicos en el rango de 0 a 5 voltios referenciados a GND con una resolución digital de 10 bits o 1024 puntos.

Las entradas analógicas están protegidas contra cortas sobretensiones superiores a 7.5 voltios.

Un ejemplo de uso es el siguiente, donde se muestra la conexión con un potenciómetro:

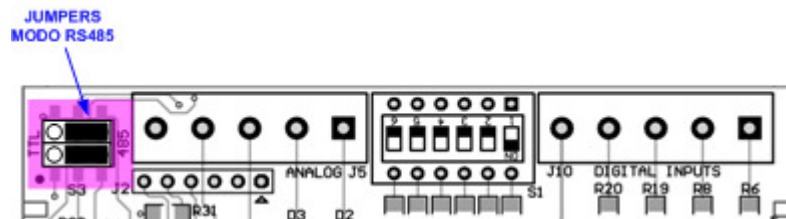


7. Comunicaciones RS485.

ArduPLCm incorpora un puerto RS485 con control automático de dirección de datos.

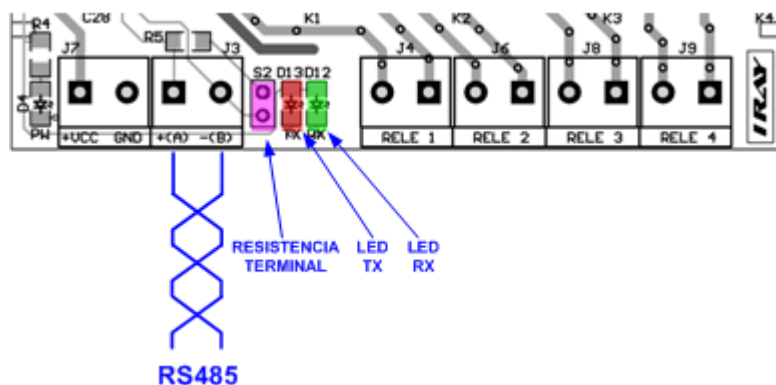
Con este puerto RS485 podemos implementar buses de comunicaciones basados en dos hilos como por ejemplo MODBUS.

Antes de utilizar el puerto debemos asegurarnos que los jumpers S3 están colocados correctamente en el modo RS485.



Si es necesario, puede activar o desactivar la resistencia terminal del bus con el jumper S2. Esta resistencia tiene un valor típico de 120 Ohm.

Puede observar la actividad de las comunicaciones del puerto con los leds TX y RX.



CONSEJO: puede utilizar el dipswitch para direccionar varios ArduPLCm dentro de una red RS485.

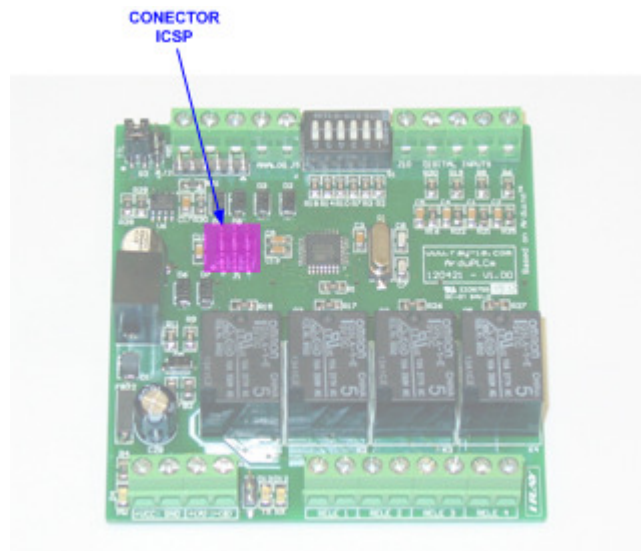
8. Programación.

Puede programar fácilmente ArduPLCm de dos formas:

- Mediante un programador externo a través del conector ICSP. Esta es la forma nativa de programar los microcontroladores AVR de Atmel. El programador más utilizado para este fin es el AVR MKII:

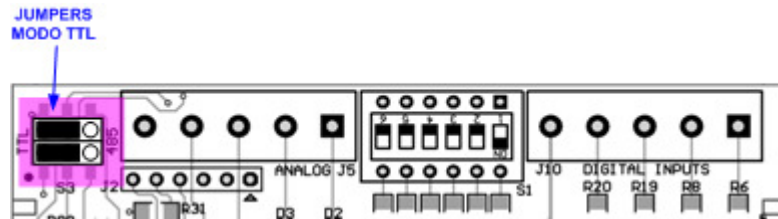


Para programar el microcontrolador con este programador tenemos que acceder a la placa ya sea quitando la tapa superior o inferior de la caja DIN.



- Mediante el puerto serie TTL disponible y el bootloader cargado en el microcontrolador.

ArduPLCm viene de fábrica con el bootloader de Arduino Duemilanove cargado por lo que solo necesitamos un puerto serie TTL en nuestro ordenador para descargar programas desde el IDE de Arduino. Antes de cargar un programa asegúrese que los jumpers S3 están colocados en modo TTL:

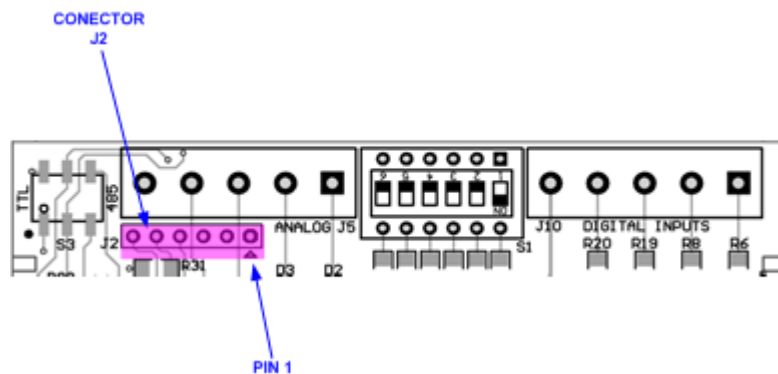


Existen cables USB comerciales con salida serie TTL. ArduPLCm es compatible con el cable comercial FTDI modelo TTL-232R-5V:



El pinout del conector J2 es el siguiente:

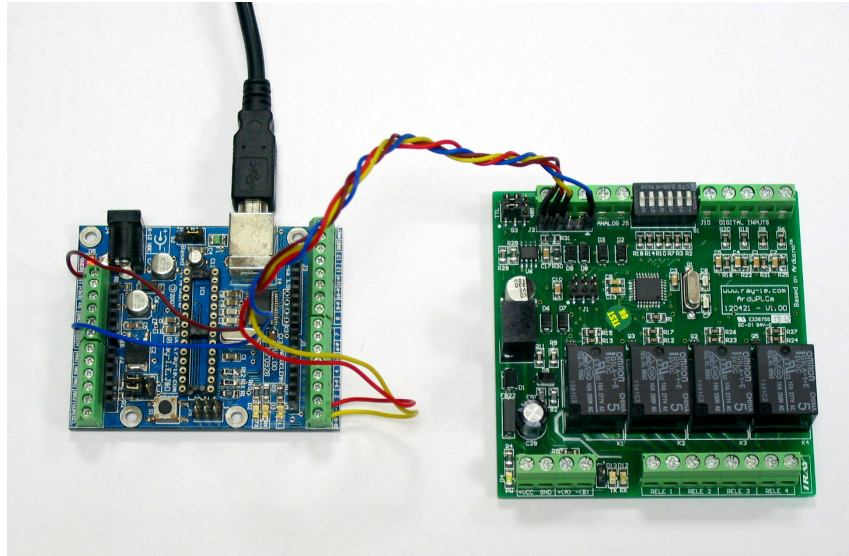
PIN	SEÑAL	COLOR
1	GND	NEGRO
2	CTS	MARRON
3	NO UTILIZADO	ROJO
4	TXD (RX del microcontrolador)	NARANJA
5	RXD (TX del microcontrolador)	AMARILLO
6	RTS (RESET del microcontrolador)	VERDE



Si no disponemos de un cable de este tipo podemos usar como alternativa una placa Arduino o compatible que incorpore puerto USB. Para convertir un Arduino en una pasarela USB a puerto serie TTL solo tenemos que extraer el microcontrolador de su zócalo y hacer las siguientes conexiones:

Arduino sin microcontrolador	Conector J2 de ArduPLCm
PIN 0	4 (TXD)
PIN 1	5 (RXD)
RST	6 (RTS)
GND	1 (GND)

Puede ver el montaje en la siguiente ilustración. Aquí se muestra con una placa Arduclema:



9. Alimentación.

ArduPLCm incorpora un regulador de tipo conmutado para obtener los 5 voltios necesarios para su funcionamiento. De esta forma aumentamos la eficiencia energética del este producto.

En las versiones convencionales de Arduino, el regulador lineal de tensión desperdicia la potencia sobrante en disipación de calor, lo cual provoca un sobrecalentamiento excesivo en la placa si ésta es alimentada a tensiones elevadas. Con el regulador conmutado esto no sucede ya que al aumentar la tensión de alimentación, la corriente de alimentación disminuye manteniendo constante la potencia y por lo tanto sin desperdicio energético y sin calentamientos.

10. Librería ArduPLCm para Arduino 1.0.

Puede instalar la librería ArduPLCm copiando el directorio “ArduPLCm” dentro de la carpeta “libraries” de Arduino 1.0.

Las funciones soportadas son:

- **leeDIPSW():** lee el estado del dipswitch en formato binario. Devuelve un entero entre 0 y 63.
- **leeAIN(canal):** lee una entrada analógica en formato nativo de 10 bits. Devuelve un número entero de 0 a 1023. “canal” es el canal analógico desde AIN1 hasta AIN4.
- **leeVoltiosAIN(canal):** lee una entrada analógica en voltios. Devuelve un número flotante entre 0 y 5. “canal” es el canal analógico desde AIN1 hasta AIN4.
- **leeDIN(canal):** lee una entrada digital. Devuelve 0 o 1 dependiendo del estado. “canal” es el canal digital desde DIN1 hasta DIN4.
- **releOn(canal):** activa un relé. “canal” es el relé a activar desde RELE1 hasta RELE4.
- **releOff(canal):** desactiva un relé. “canal” es el relé a desactivar desde RELE1 hasta RELE4.

11. Especificaciones técnicas.

Rango de alimentación:	6.5 a 30 VDC
Protección de alimentación:	inversión de polaridad
Protecciones de sobretensión en:	entradas analógicas y puerto RS485
Consumo a 24VDC en reposo:	10mA (0.24W)
Consumo a 24VDC máximo:	80mA (1.92W)
Rango entradas analógicas:	0 a 5 VDC
Rango entradas digitales:	0 a 5VDC
Máxima corriente salidas relé:	10A
Máxima tensión salidas relé:	250VAC o 30VDC
Máxima potencia salidas relé:	2500VA, 300W
Puerto RS485:	No aislado, ¼ unidad de carga, protección ESD +/- 15Kv, control automático de dirección de datos
Temperatura de funcionamiento:	-40 a 85 °C
Ancho:	87 mm
Alto:	90 mm
Fondo:	58 mm
Peso:	150 g.