



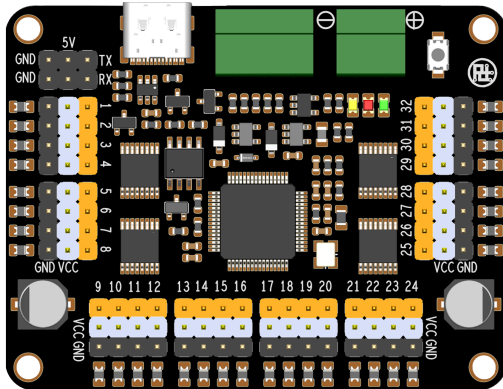
Async Servo Motor Controller
Instructions for use
伺服電機异步控制器使用說明

Ver 1.0

參數對比：

硬體	异步 32 通道	同步 32 通道
工作電壓	5V	
伺服電機輸入電壓	根據伺服電機為準	
CPU	32 位	
串列傳輸速率 (USB)	115200	
串列傳輸速率 (藍牙、UART)	4800、9600、19200、38400、57600、115200	
可同時操控伺服電機數	32 通道	
儲存 Flash 容量	無	16M
最大動作組數	無	255 組
控制精度	1us	
頻率	50Hz	
伺服電機信號隔離	有	有
限流保護	有	無
並聯支持	支持 (UART/RS485)	不支持
通道獨立速度	支持	不支持
3D 展示效果介面	全部	全部
指示燈	1.CPU 電源指示燈 (紅) 2.伺服電機指示燈 (綠) 3.低壓警報 (黃)	1.CPU 電源指示燈 (紅) 2.伺服電機指示燈 (綠) 3.無線手柄模式 (黃)
板子尺寸	66mm X 50.5mm	64mm X 45mm
通訊協定	RTrobot、Modbus RTU(UART/RS485)	RTrobot
硬體介面	UART、USB、RS485	UART、USB
支援系統	Windows 10 以上, Mac OS 10.8 以上, Linux (glibc-2.35 以上)	
低壓警報	支持	
伺服電機初始值	默認 1500	
支援伺服電機類型	9G~55G	
離線獨立運行	不支持	支持
線上作業支持	C51、Arduino、ARM、MSP、DSP、藍牙、Wi-Fi、電腦	
外設支援	無	PS2 手柄、電位器

32 通道：



指令：

通訊協定：

串口通訊	串列傳輸速率	校驗位	數據位元	停止位
TTL	115200（默認）	無	8	1

指令格式：

名稱	指令	說明
控制單個伺服電機	#1P1500T1000\r\n	數據 1 為伺服電機編號 資料 1500 為伺服電機角度值，範圍為 500-2500 資料 1000 為伺服電機執行速度，範圍是 1-9999
控制多個伺服電機	#1P1500T789#2P1234T1000\r\n	數據 1、2 為伺服電機編號 資料 1500 為伺服電機 1 角度值，範圍為 500-2500 資料 789 為伺服電機 1 執行速度，範圍是 1-9999 資料 1234 為伺服電機 2 角度值，範圍為 500-2500 資料 1000 為伺服電機 2 執行速度，範圍是 1-9999

注意：“\r\n”轉化為 16 進制為“0X0D 0X0A”；所有指令都為 ASCII 碼。

“0x0D” == “\r” == “CR”

“0x0A” == “\n” == “LF”

提示：如果程式設計中使用的函數或者軟體有“\r\n”功能則末尾無需再加。當指令執行完成後控制器不會回饋。

系統指令：

指令	說明
~RESTART	重新啟動控制器
~STOP	停止通道，（注意：停止後不可恢復，只能發送新的指令重新執行） 參數： =ALL（停止所有通道） =15（停止 15 號通道） 例子： ~STOP=15\r\n
~GET+POS	獲取通道當前值 參數： =15（獲取 15 號通道當前值） 例子： ~GET+POS=15\r\n 回饋： ~POS15:1500\r\n

Modbus RTU 協議：

寄存器表：

寄存器位址	通道號	功能
0x00	Null	停止通道運行 (寫入對應序號則停止對應通道，寫入 0x00FF 停止所有通道)
0x01	1	伺服電機目標位置值(uint16)
0x02	1	伺服電機運行速度(uint16)
0x03	2	伺服電機目標位置值(uint16)
0x04	2	伺服電機運行速度(uint16)
0x05	3	伺服電機目標位置值(uint16)
0x06	3	伺服電機運行速度(uint16)
0x07	4	伺服電機目標位置值(uint16)
0x08	4	伺服電機運行速度(uint16)
0x09	5	伺服電機目標位置值(uint16)
0x0A	5	伺服電機運行速度(uint16)
0x0B	6	伺服電機目標位置值(uint16)
0x0C	6	伺服電機運行速度(uint16)
0x0D	7	伺服電機目標位置值(uint16)
0x0E	7	伺服電機運行速度(uint16)
0x0F	8	伺服電機目標位置值(uint16)
0x10	8	伺服電機運行速度(uint16)
0x11	9	伺服電機目標位置值(uint16)
0x12	9	伺服電機運行速度(uint16)
0x13	10	伺服電機目標位置值(uint16)
0x14	10	伺服電機運行速度(uint16)
0x15	11	伺服電機目標位置值(uint16)
0x16	11	伺服電機運行速度(uint16)
0x17	12	伺服電機目標位置值(uint16)
0x18	12	伺服電機運行速度(uint16)
0x19	13	伺服電機目標位置值(uint16)
0x1A	13	伺服電機運行速度(uint16)
0x1B	14	伺服電機目標位置值(uint16)
0x1C	14	伺服電機運行速度(uint16)
0x1D	15	伺服電機目標位置值(uint16)
0x1E	15	伺服電機運行速度(uint16)

0x1F	16	伺服電機目標位置值(uint16)
0x20	16	伺服電機運行速度(uint16)
0x21	17	伺服電機目標位置值(uint16)
0x22	17	伺服電機運行速度(uint16)
0x23	18	伺服電機目標位置值(uint16)
0x24	18	伺服電機運行速度(uint16)
0x25	19	伺服電機目標位置值(uint16)
0x26	19	伺服電機運行速度(uint16)
0x27	20	伺服電機目標位置值(uint16)
0x28	20	伺服電機運行速度(uint16)
0x29	21	伺服電機目標位置值(uint16)
0x2A	21	伺服電機運行速度(uint16)
0x2B	22	伺服電機目標位置值(uint16)
0x2C	22	伺服電機運行速度(uint16)
0x2D	23	伺服電機目標位置值(uint16)
0x2E	23	伺服電機運行速度(uint16)
0x2F	24	伺服電機目標位置值(uint16)
0x30	24	伺服電機運行速度(uint16)
0x31	25	伺服電機目標位置值(uint16)
0x32	25	伺服電機運行速度(uint16)
0x33	26	伺服電機目標位置值(uint16)
0x34	26	伺服電機運行速度(uint16)
0x35	27	伺服電機目標位置值(uint16)
0x36	27	伺服電機運行速度(uint16)
0x37	28	伺服電機目標位置值(uint16)
0x38	28	伺服電機運行速度(uint16)
0x39	29	伺服電機目標位置值(uint16)
0x3A	29	伺服電機運行速度(uint16)
0x3B	30	伺服電機目標位置值(uint16)
0x3C	30	伺服電機運行速度(uint16)
0x3D	31	伺服電機目標位置值(uint16)
0x3E	31	伺服電機運行速度(uint16)
0x3F	32	伺服電機目標位置值(uint16)
0x40	32	伺服電機運行速度(uint16)

停止示例：

設備號	功能位	起始位高	起始位低	寄存器數 量高	寄存器數 量低	數據長度	序號高	序號低	CRC 高	CRC 低
-----	-----	------	------	------------	------------	------	-----	-----	-------	-------

停止所有通道運行：

0x01,0x10,0x00,0x00,0x00,0x01,0x02,0x00,0xFF,0xE6,0x10

停止通道 7 運行：

0x01,0x10,0x00,0x00,0x00,0x01,0x02,0x00,0x07,0xE7,0x92

停止通道 16 運行：

0x01,0x10,0x00,0x00,0x00,0x01,0x02,0x00,0x10,0xA7,0x9C

讀取通道值示例：

設備號	功能位	起始位高	起始位低	寄存器數量高	寄存器數量低	CRC 高	CRC 低
-----	-----	------	------	--------	--------	-------	-------

讀取通道 5 的當前位置值(寄存器位址:0x09)：

0x01,0x04,0x00,0x09,0x00,0x01,0xE1,0xC8

讀取通道 32 的當前位置值(寄存器地址:0x3F)：

0x01,0x04,0x00,0x3F,0x00,0x01,0x01,0xC6

寫入通道值示例：

設備號	功能位	起始位 高	起始位 低	寄存器 數量高	寄存器 數量低	數據長 度	通道位 置高	通道位 置低	速度高	速度低	CRC 高	CRC 低
-----	-----	----------	----------	------------	------------	----------	-----------	-----------	-----	-----	-------	-------

寫入通道 1 的目標值為 1734(0x06C6)與速度值 5478(0x1566) (寄存器位址:0x01, 0x02):

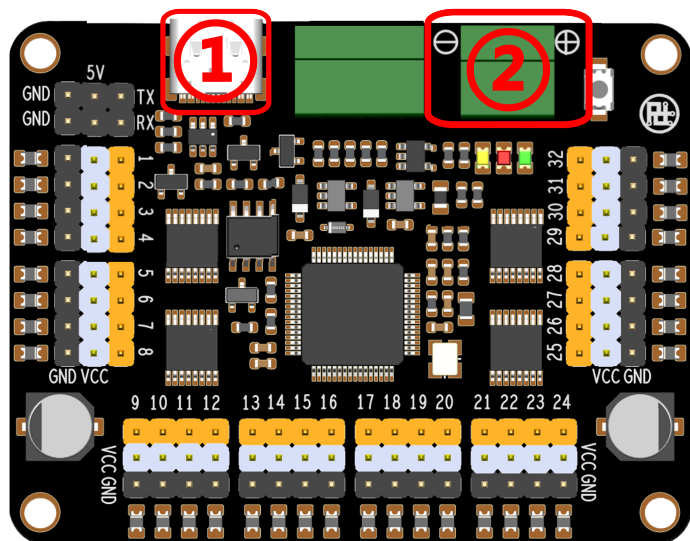
0x01,0x10,0x00,0x01,0x00,0x02,0x04,0x06,0xC6,0x15,0x66,0x5C,0x6C

寫入通道 31 的目標值為 534(0x0216)與速度值 478(寄存器位址:0x01DE):

0x01,0x10,0x00,3D,0x00,0x02,0x04,0x02,0x16,0x01,0xDE,0x50,0x96

接線方法:

一、電源接入方法，如圖一位置：



圖一

VCC：為伺服電機電源 VCC 輸入口，可接入 4.2V~12V 電源，請接入電源正極。

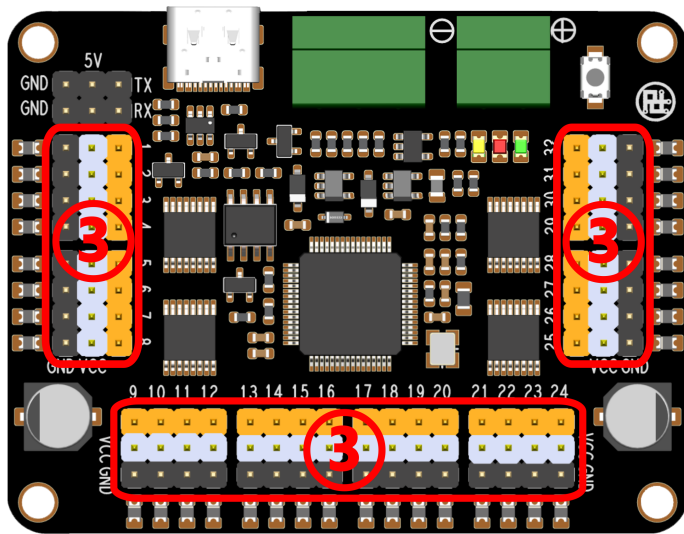
注意：控制器的 VCC 介面是伺服電機的電源輸入口。VCC 介面應該根據伺服電機的要求選擇，例如 1 個伺服電機需要 6V 電壓峰值 2A 電流，10 個伺服電機電源則應該選擇為 6V 電壓，10A 電流的電源。

GND：為伺服電機控制器的整體 GND，可接入伺服電機電源 GND 或者是 CPU 電源 GND，請接入電源負極。

USB (1)：同為伺服電機控制器的 CPU 電源輸入口和資料通信口。

USB 介面只支援 RTrobot 協定，不支援 Modbus 協定。

二、伺服電機接線方法，如圖二位置：



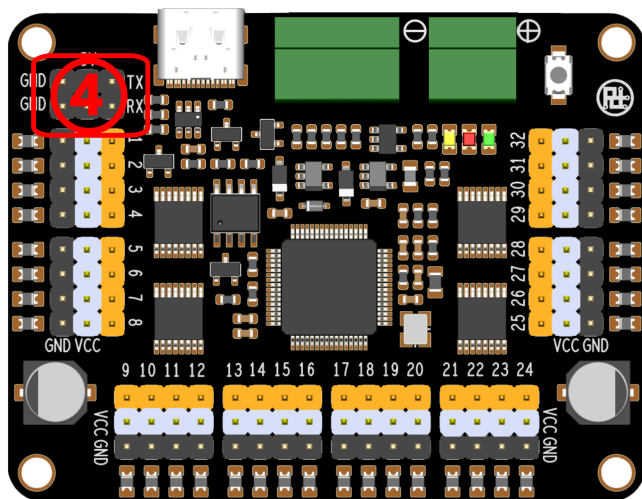
圖二

黃色排針：為伺服電機 I/O 接入口，一般是伺服電機的黃色或者是土黃色。

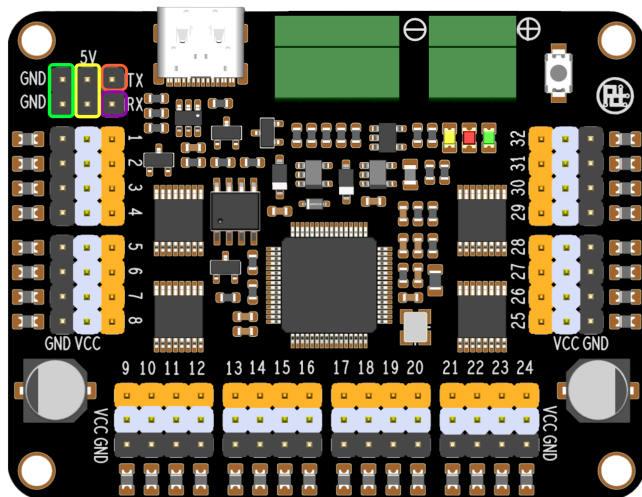
白色排針：為伺服電機 VCC 接入口，一般是伺服電機的白色、紅色或者是暗紅色。

黑色排針：為伺服電機 GND 接入口，一般是伺服電機的褐色或者是黑色。

三、UART 接線方法，如圖三位置，配合圖四閱讀：



圖三



圖四

綠色圈位置：為伺服電機控制器 CPU 電源入口 GND。

黃色圈位置：為伺服電機控制器 CPU 電源入口 VCC，僅可接 5V。

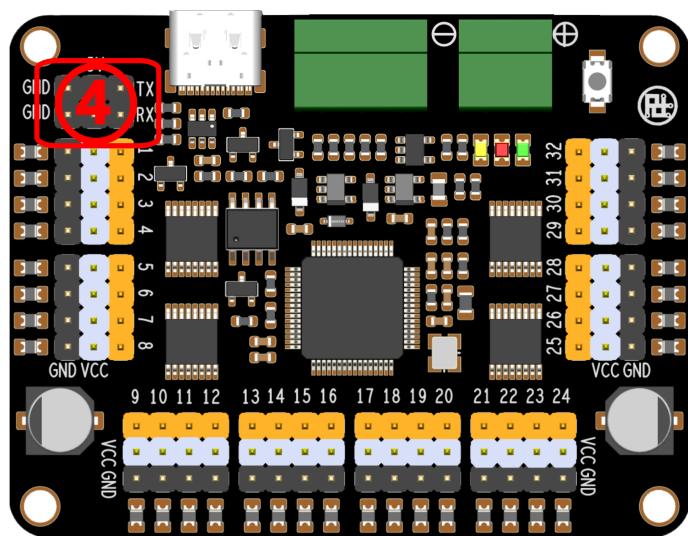
紫色圈位置：為伺服電機控制器 UART 的 RX 口，一般接入其他 UART 設備的 TX 口。

橙色圈位置：為伺服電機控制器 UART 的 TX 口，一般接入其他 UART 設備的 RX 口。

UART 可以支援兩種協定，通過上位機可修改為 Modbus 協議。

UART 介面不可以並聯。

四、藍牙、Wi-Fi 感測器接線方法，如圖五位置：



圖五

圖五位置使用 4 根杜邦線與藍牙、Wi-Fi 模組連結，5V-VCC，GND-GND，RX-TX，TX-RX。

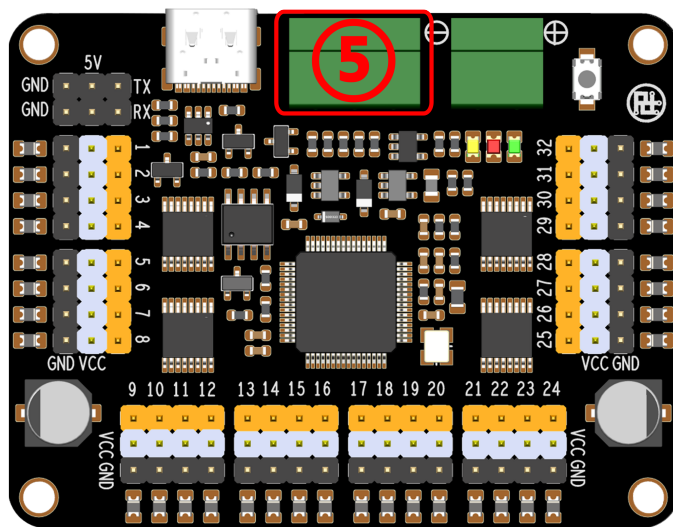
使用手機與藍牙模組配對，安裝打開手機控制軟體即可進行控制。

安裝打開手機控制軟體，填寫 Wi-Fi 模組設置的 TCP 位址即可進行控制。

溫馨提示：在實際使用手機遙控前，應把藍牙、Wi-Fi 模組接入電腦，使用串口調試軟體查看是否能接收到相應指令。

首次使用手機軟體的時候需要輸入驗證碼，驗證碼為：**RTrobot**（注意大小寫區分）

五、RS485 介面如圖六：

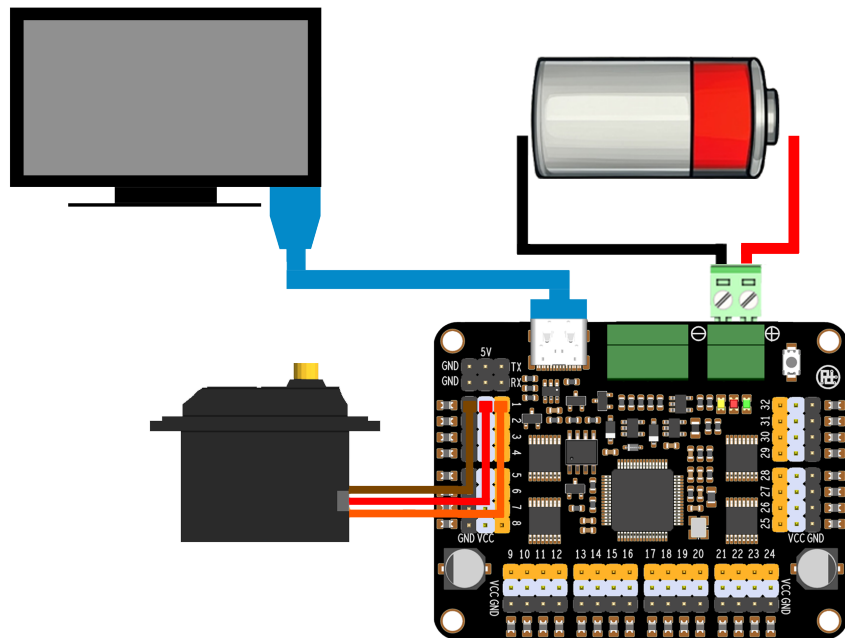


圖六

圖六位置為 RS485 2-Wire 介面，可以支援兩種協定，通過上位機可修改為 Modbus 協議。RS485 介面可以並聯多個控制器，但需要先修改為 Modbus 協定，如果設置為 RTrobot 協議無法並聯。

整體接線例示:

一、使用電腦控制:



圖七

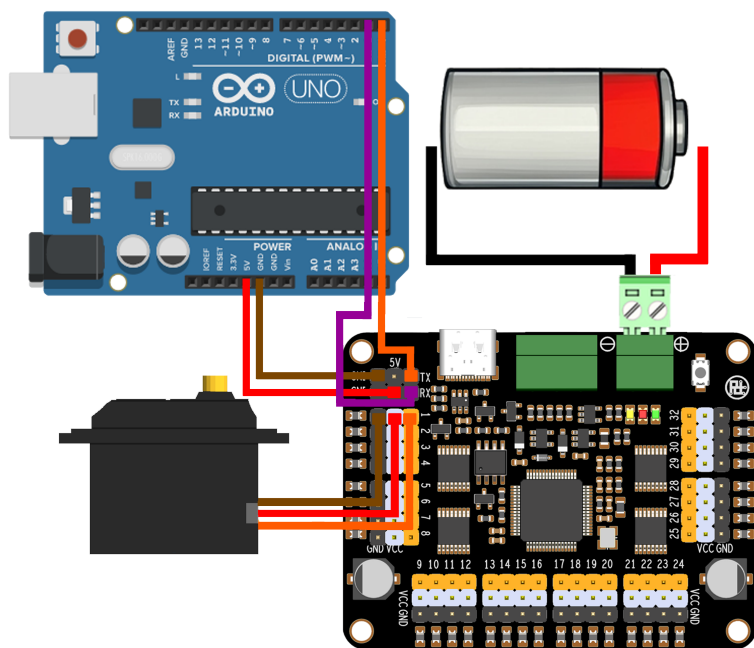
先把伺服電機與伺服電機的電源接入，然後使用 USB 線把電腦與伺服電機控制器連結在一起。

伺服電機的電源接入請參考**接線方法一**

溫馨提示： 伺服電機電源應根據伺服電機要求選擇。

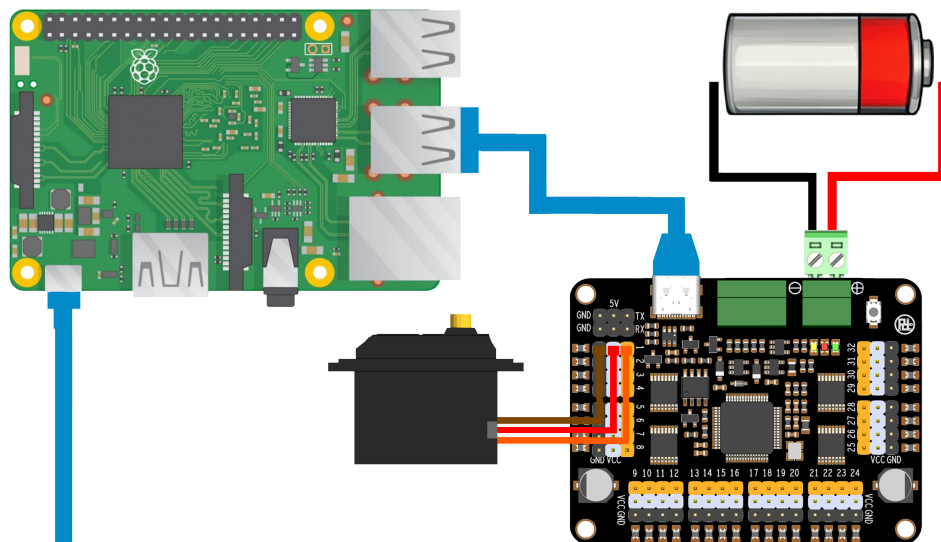
二、使用 MCU 控制：

這裡以 Arduino uno 為例子，其他 MCU 參考此處。控制器 5V 接 Arduino uno 的 5V，控制器 GND 接 Arduino uno 的 GND，控制器 TX 接 Arduino uno 的 RX，控制器 RX 接 Arduino uno 的 TX。如圖八。



圖八

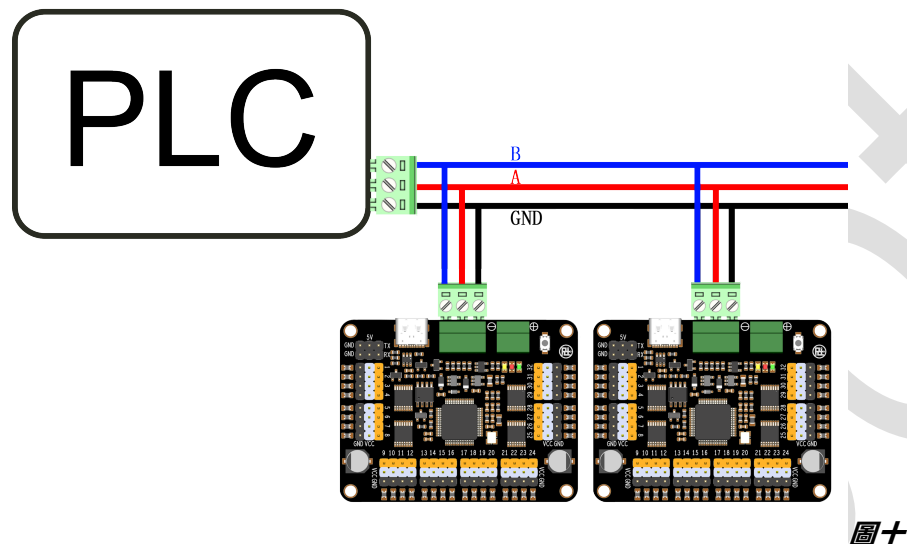
Raspberry Pi 用戶可以使用 USB 線將控制器連接到 Raspberry Pi 上，把它當作一台電腦。



圖九

二、使用 RS485 控制：

RS485 2-Wire 介面可以設置為 Modbus 協定，使用 Modbus 協定可以並聯多個控制器以達到更多伺服電機的控制。在使用前，需要在上位機中設置 Modbus 的從位址。



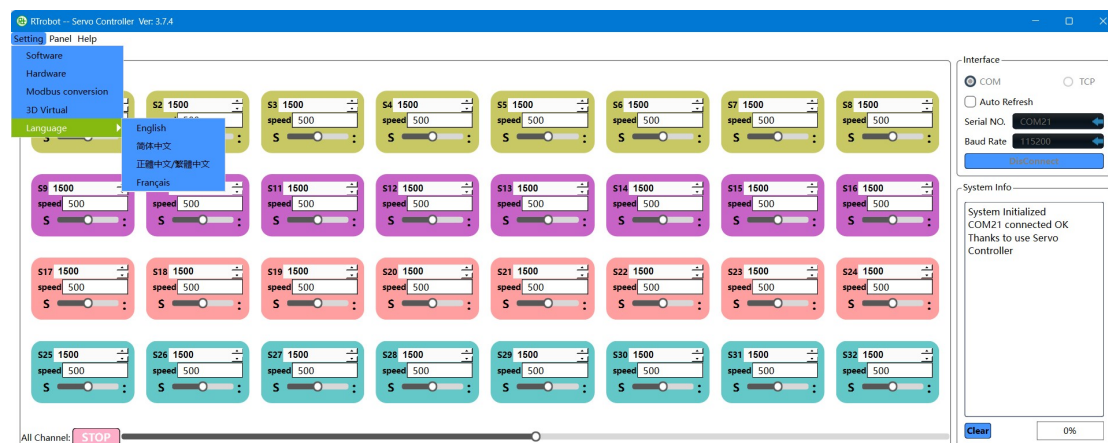
圖十

軟體操作方法:

注意：需打開串口才能使用軟體的全部功能

一、語言設置:

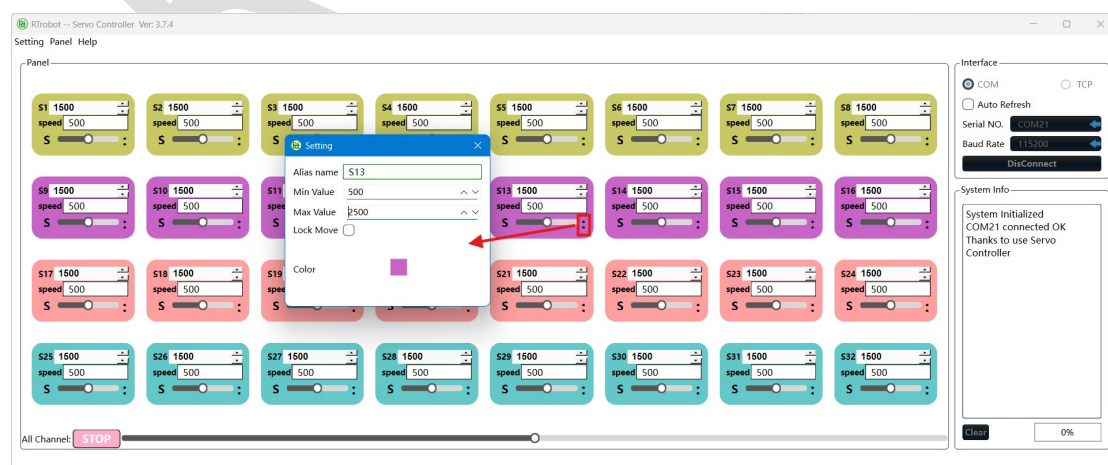
點擊“Setting”->“Language”選擇語言（簡體/繁體/English）。



圖十一

二、軟體設置:

点击伺服電機窗口的“⚙️”，可以單獨對每個伺服電機命名與設置最大最小值、顏色和鎖定位置。



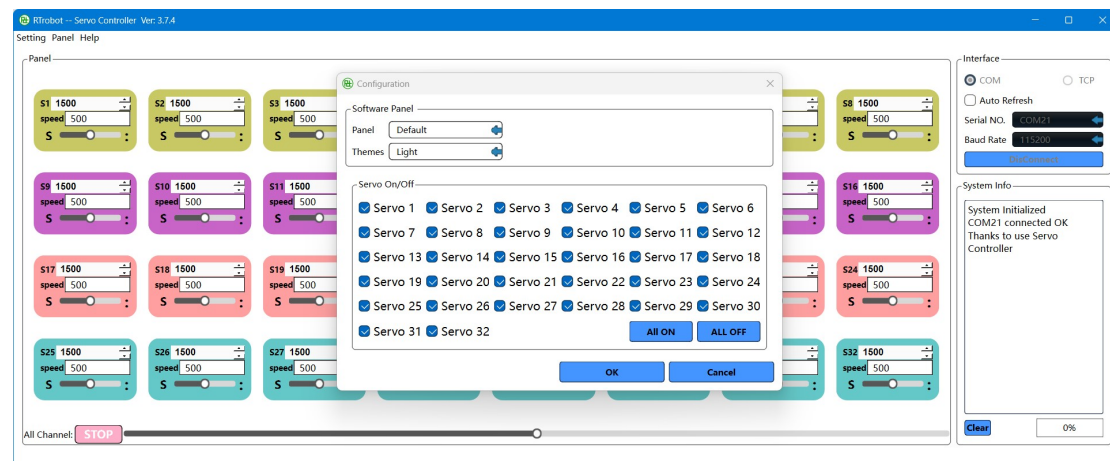
圖十二

點擊“Setting”->“Software”可對軟體進行設置，如圖十三。

Software Panel：設置軟體控制台。

Servo On/Off：顯示需要控制的伺服電機序號。

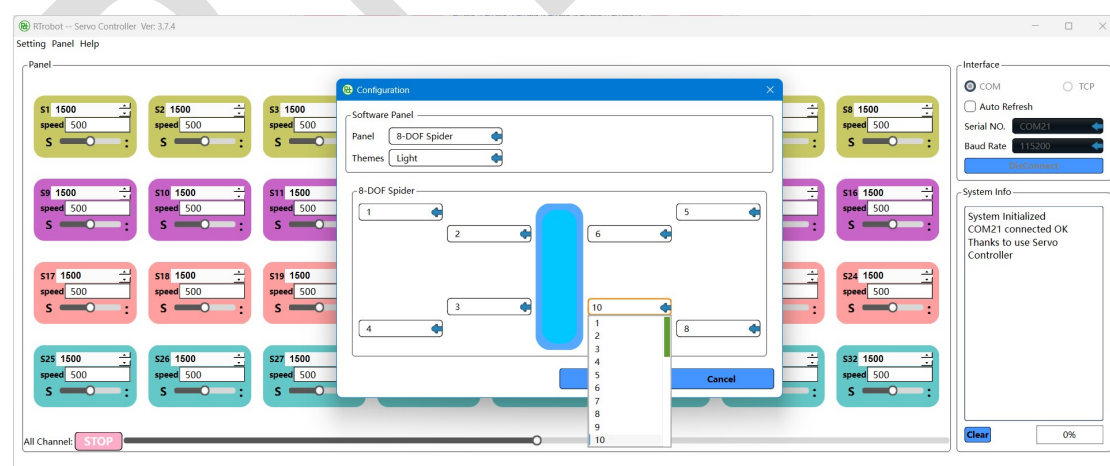
軟體設置完成後點擊“OK”會自動重新開機軟體。



圖十三

Software Panel 選擇介面後可指定每個位置的伺服電機序號，如圖十四。

注意：如果存在重複序號的伺服電機，則無法保存。



圖十四

三、控制器設置：

點擊“Setting”->“Hardware”可對控制器進行設置，如圖十五。

Servo initial value：設置每個伺服電機上電啟動的初始值。

Servo Deviation Value：設置每個伺服電機的偏差值（有效值：**-99~99**），如圖十六。

UART Baud Rate：設置圖五中④位置的 UART 串列傳輸速率。

RS485 Baud Rate：設置圖六中⑤位置的 RS485 串列傳輸速率。

Protocol：設置 UART/RS485 的協議。

Modbus：Modbus 的從地址。

注意：設置完畢後記住點擊 Apply，等待設置完畢。每次設置後控制器需要重新開機才會生效。

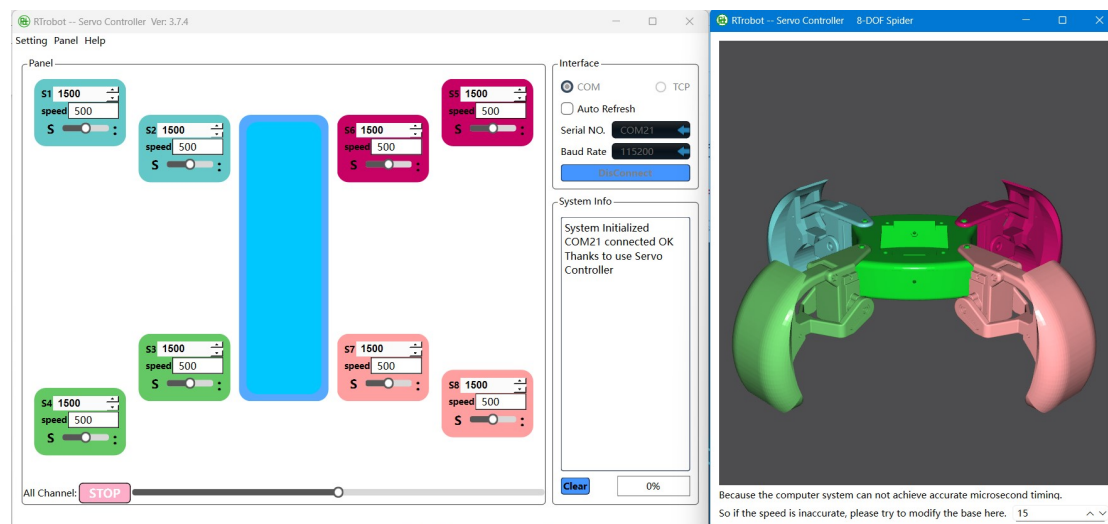
圖十五

圖十六

四、3D 展示效果：

點擊“Setting”->“3D Virtual”可展示 3D 效果介面

注意：需要先在“Software”設置頁選擇好控制介面後打開串口才能展示 3D 效果介面



圖十七

五、軟體控制：

1. 選擇一種合適自己的接線方式，並使用 USB 線與電腦連接好。
2. 安裝控制器的驅動（Servos Controller Drive.exe）（Windows10 以上無需安裝驅動）。

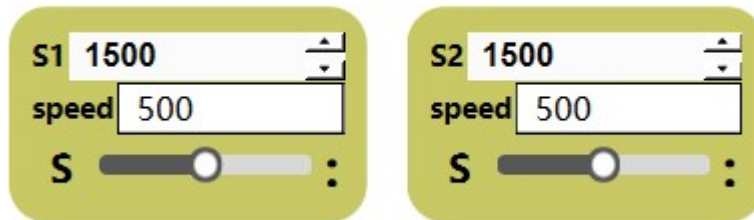
注意：如提示無數位簽章無法安裝，則把系統禁用驅動數位簽章方式啟動電腦系統再次運行安裝。

3. 打開軟體“ServoController.exe”。
4. 選擇好串口，並且打開串口。如果使用 Wi-Fi 模式，選擇 TCP，填寫好 Wi-Fi 模組 TCP 伺服器設定的 TCP 位址與埠。

注意：只有使用 USB 線連接電腦時才能使用全部功能。

單個伺服電機操作：

如圖十八所示，拖動或者是填寫都可以改變伺服電機角度值，每個通道都可以設置不同的速度，範圍是：1~9999。"S"為停止按鈕，可以停止當前通道。



圖十八

多個伺服電機操作：



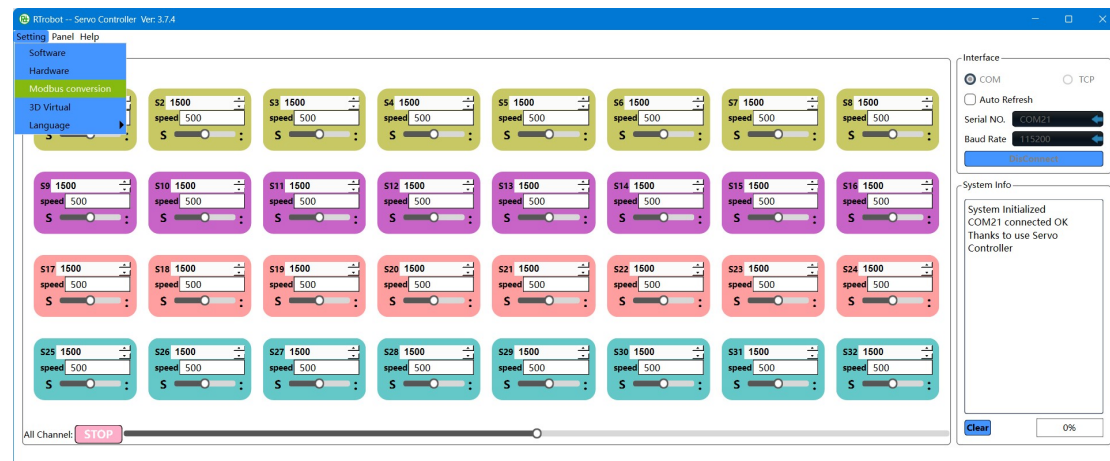
圖十九

"STOP"為全部通道的停止按鍵，對應指令為"**~STOP=ALL\r\n**"。拖動條可以全通道控制，速度為每個通道獨立的"speed"值。

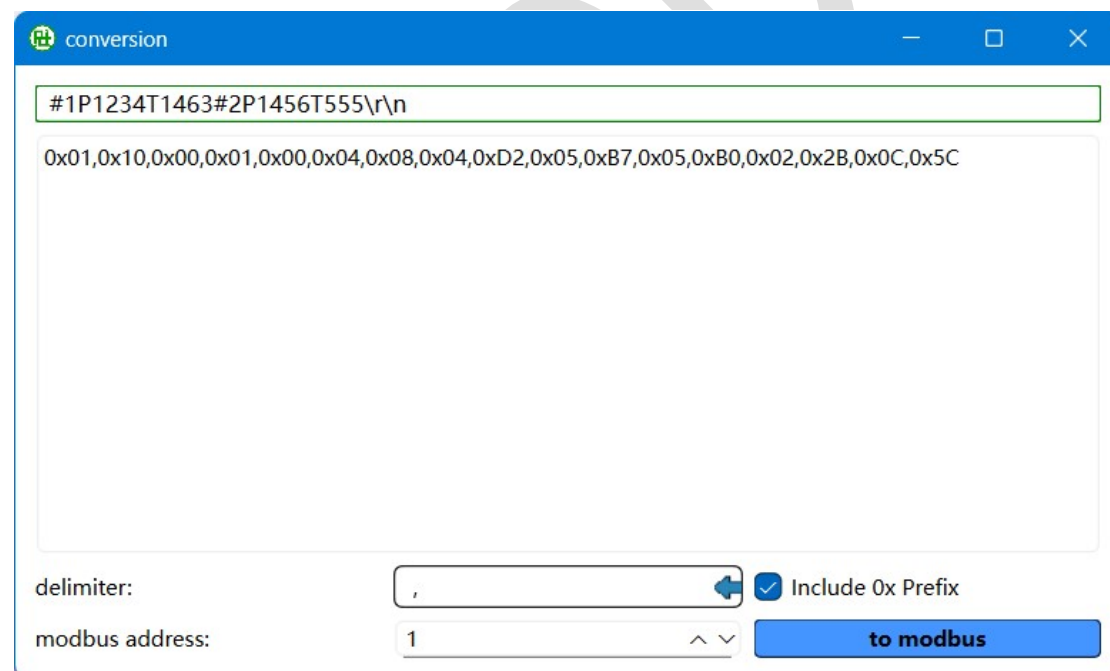
RTrobot 转 Modbus 协议:

为了方便使用 Modbus 协议，免于查表快速输入 Modbus 指令，可以使用软件自带的功能

"Modbus conversion"进行快速转换。



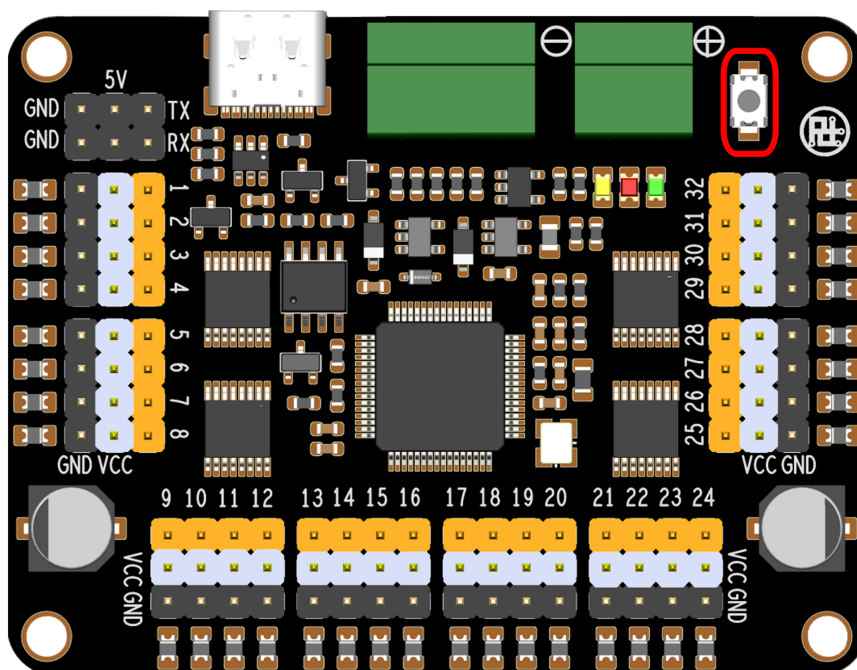
图二十



图二十一

六、升級固件方法

1. 在 "<http://www.rtrobot.org/software/>" 網站下載最新的上位機軟體
2. 打開最新上位機軟體
3. 按住伺服電機控制器上的按鍵不放的狀態下連結 USB 資料線後鬆開按鍵

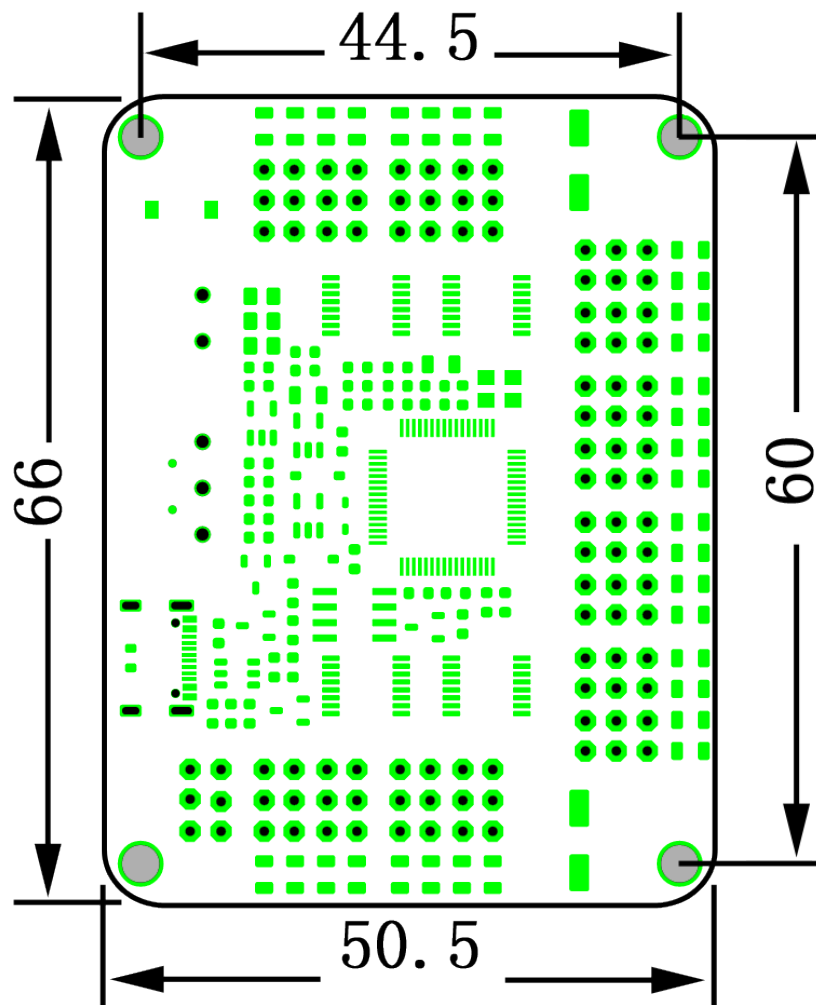


圖二十二

4. 打開最新上位機軟體的串口，這時固件開始升級，升級完後會提示 "Update Success, Restart the controller, Please. "
5. 重啟伺服電機控制器。
6. 如已是最新固件的話會提示 "ERROR Don't need Update!"

尺寸圖：

32:



十、關於：

感謝使用 RTrobot 出品的伺服電機控制器，在使用中對控制器有任何疑問需要諮詢可

EMAIL:admin@rtrobot.org。