

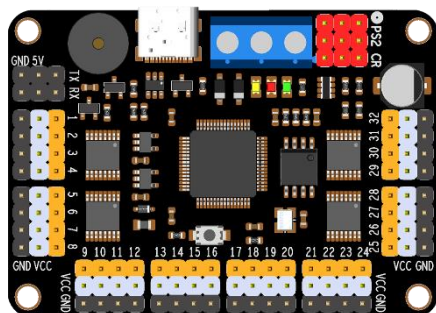
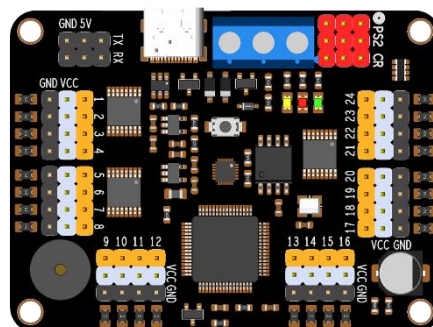
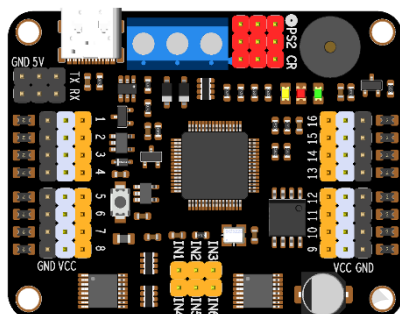


Servo Motor Controller Instructions
for use
舵机控制器使用说明

Ver 3.15

参数：

硬件	32	24	16
工作电压	5V		
舵机输入电压	根据舵机为准		
CPU	32位		
波特率 (USB)	115200		
波特率 (蓝牙、UART)	4800、9600、19200、38400、57600、115200		
储存 Flash 容量	16M		
可同时操控舵机数	32 路	24 路	16 路
最大动作组数	255组		
控制精度	1us		
频率	50Hz		
舵机信号隔离	有	有	有
限流保护	无	有	有
MPU6500	无	有	无
外部传感器支持	无	无	有
3D 展示效果界面	全部	全部	部分
指示灯	1.CPU 电源指示灯 (红) 2.舵机指示灯 (绿) 3.无线手柄模式 (黄)		
板子尺寸	64mm X 45mm	64mm X 47.5mm	58.5mm X 45mm
通讯协议	UART	UART	UART
支持系统	Windows 10 以上, Mac OS 10.8 以上, Linux (kernel 3.0 以上)		
低压警报	默认开		
舵机初始值	默认 1500		
支持舵机类型	9G~55G		
联机操作支持	C51、Arduino、ARM、MSP、DSP、蓝牙、WIFI、电脑		
无线手柄模式	1.单个舵机控制 2.动作组控制		

32 通道:**24 通道****16 通道:**

指令：

通讯协议：

串口通讯	波特率	校验位	数据位	停止位
TTL	9600 (默认)	无	8	1

指令格式：

名称	指令	说明
控制单个舵机	#1P1500T1000D800\r\n	数据 1 为舵机编号 数据 1500 为舵机角度值，范围为 500-2500 数据 1000 为舵机执行速度，范围是 0-9999 数据 800 为指令间隔延迟时间，范围是 0-9999
控制多个舵机	#1P1500#2P1500T1000D800\r\n	数据 1、2 为舵机编号 数据 1500 为舵机角度值，范围为 500-2500 数据 1000 为舵机执行速度，范围是 0-9999 数据 800 为指令间隔延迟时间，范围是 0-9999
运行动作组	G1F3\r\n	数据 1 为动作组编号 数据 3 为执行次数

注意：“\r\n” 转化为 16 进制为 “0X0D 0X0A”；所有指令都为 ASCII 码。

“0x0D” == “\r” == “CR”

“0x0A” == “\n” == “LF”

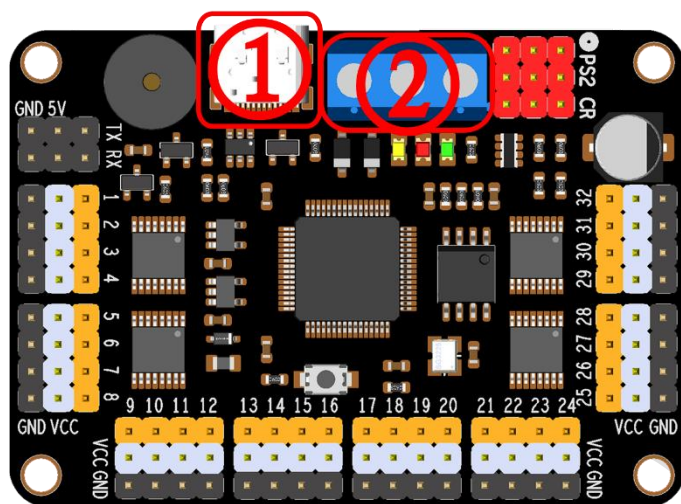
系统指令：

指令	说明
~ST	停止动作组（包括打断自动运行。 注意：停止后不可恢复，只能发送新的指令重新执行 ）
~RE	重新启动控制器

提示：如果编程中使用的函数或者软件有 “\r\n” 功能则末尾无需再加。当指令执行完成后控制器会反馈 “OK”

接线方法:

一、电源接入方法，如图一位置：



图一

VCC：为舵机电源 VCC 输入口，可接入 4.2V~9.6V 电源，请接入电源正极。

注意：控制器的 VCC 接口是舵机的电源输入口。VCC 接口应该根据舵机的要求选择，例如 1 个舵机需要 6V 电压峰值 2A 电流，10 个舵机电源则应该选择为 6V 电压，10A 电源的电源。

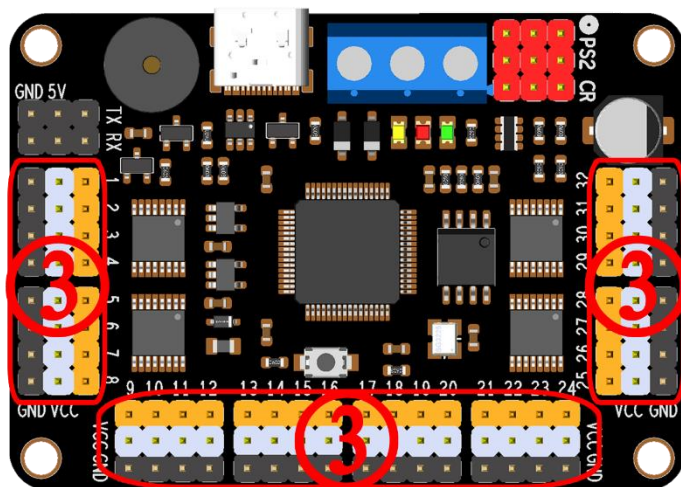
GND：为舵机控制器的整体 GND，可接入舵机电源 GND 或者是 CPU 电源 GND，请接入电源负极。

VDD：为舵机控制器的 CPU 电源输入口，供电范围 5V~9.6，请接入电源正极。

USB (①)：同为舵机控制器的 CPU 电源输入口和数据通信口。

注意：①USB 接口与②VDD 接口不能同时接入。只可选择一个作为控制器 CPU 供电端。

二、舵机接线方法，如图二位置：



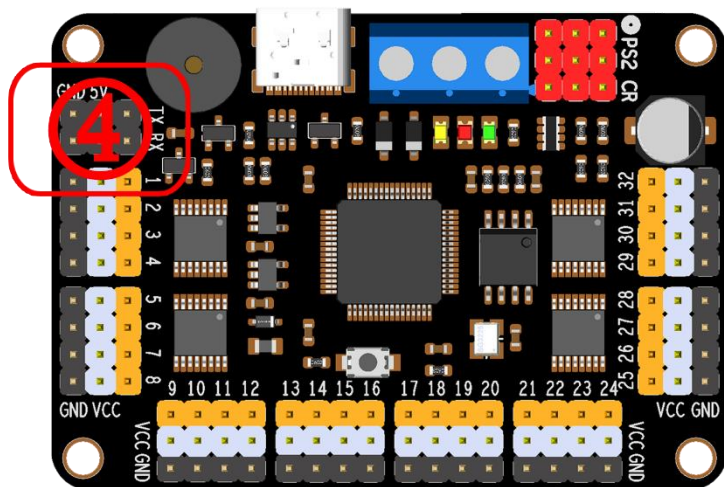
图二

黄色排针：为舵机 I/O 接入口，一般是舵机的黄色或者是土黄色。

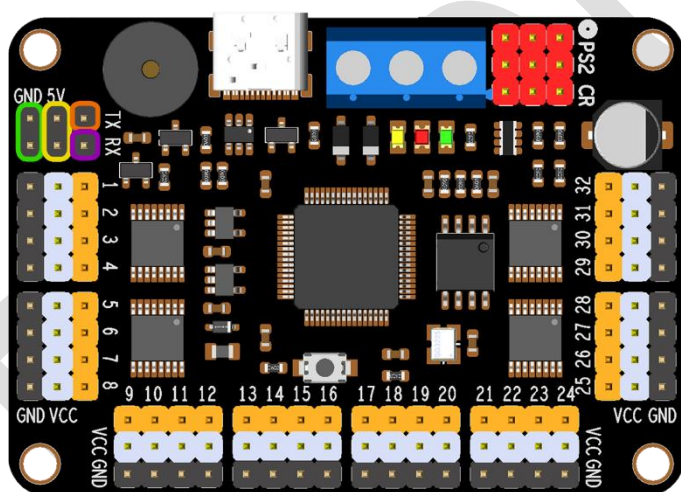
白色排针：为舵机 VCC 接入口，一般是舵机的白色、红色或者是暗红色。

黑色排针：为舵机 GND 接入口，一般是舵机的褐色或者是黑色。

三、UART 接线方法，如图三位置，配合图四阅读：



图三



图四

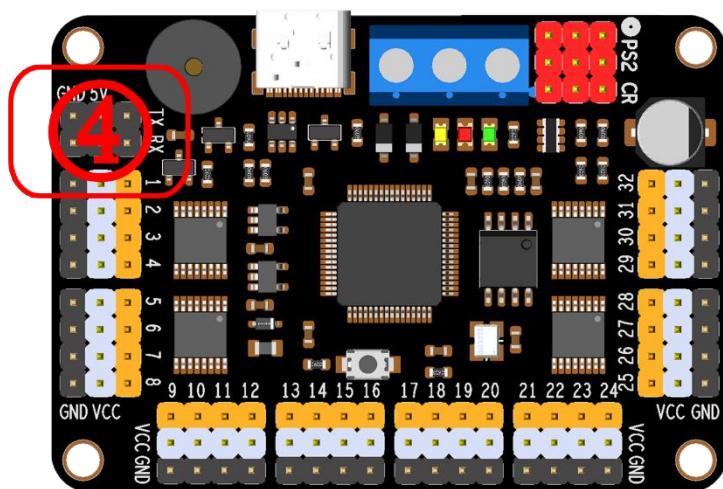
绿色圈位置：为舵机控制器 CPU 电源入口 GND。

黄色圈位置：为舵机控制器 CPU 电源入口 VCC，仅可接 5V。

紫色圈位置：为舵机控制器 UART 的 RX 口，一般接入其他 UART 设备的 TX 口。

橙色圈位置：为舵机控制器 UART 的 TX 口，一般接入其他 UART 设备的 RX 口。

四、蓝牙、WIFI 传感器接线方法，如图五位置：



图五

图五位置使用 4 根杜邦线与蓝牙、WIFI 模块链接，5V-VCC，GND-GND，RX-TX，TX-RX。

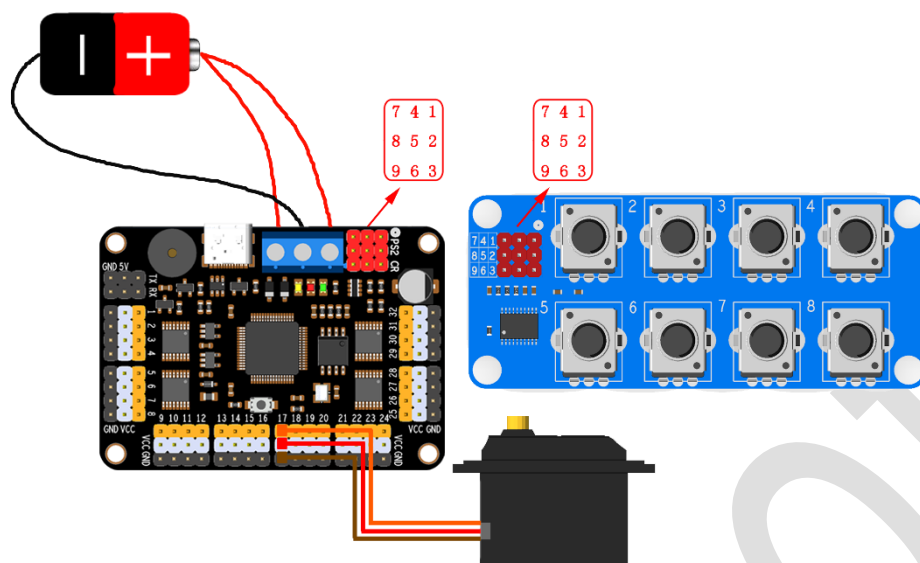
使用手机与蓝牙模块配对，安装打开手机控制软机即可进行控制。

安装打开手机控制软机，填写 WIFI 模块设置的 TCP 地址即可进行控制。

温馨提示：在实际使用手机遥控前，应把蓝牙、WIFI 模块接入电脑，使用串口调试软件查看是否能接收到相应指令。

首次使用手机软件的时候需要输入验证码，验证码为：RTrobot (注意大小写区分)

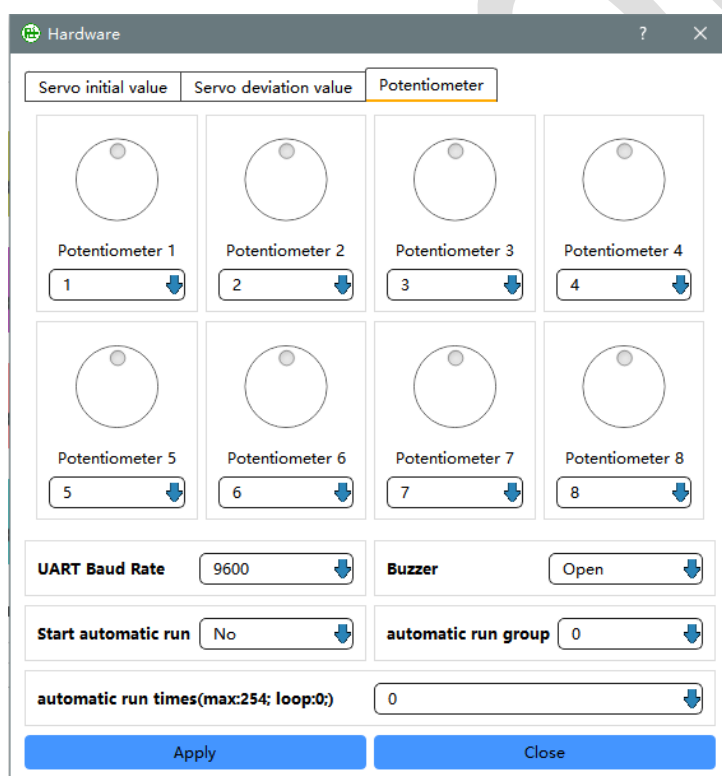
五、电位器接线方法，如图六：



图六

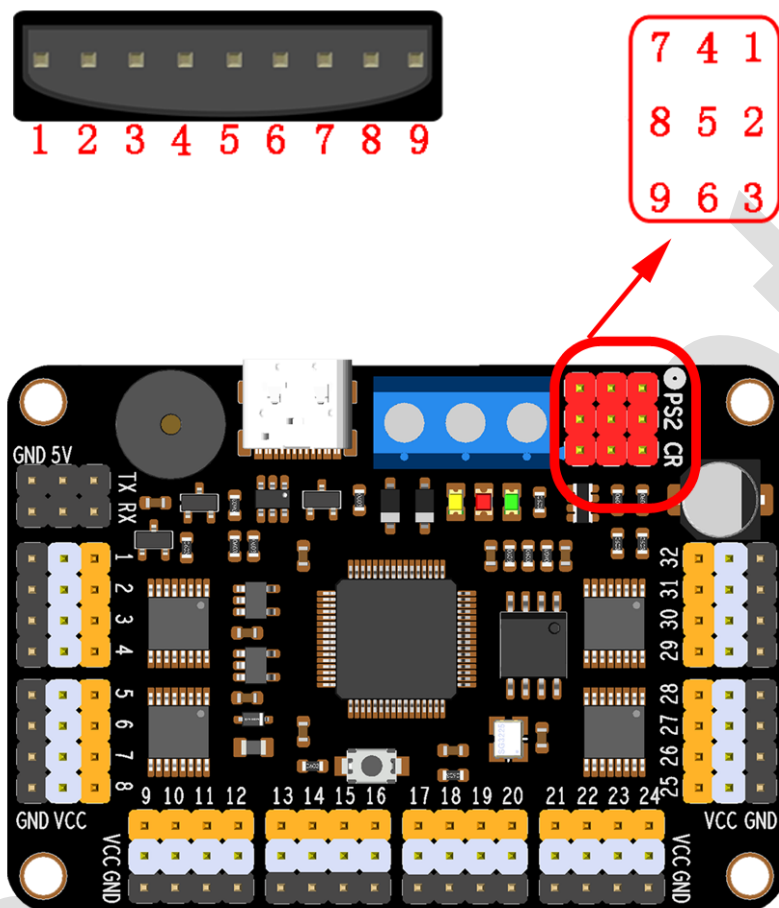
使用电位器模块与舵机控制器链接在一起，如图六，1 - 1、2 - 2、3 - 3.....

每个电位器都可以通过上位机软件设置控制哪一通道舵机 "Setting" -> "Hardware".



图七

六、无线手柄接线方法，如图八：



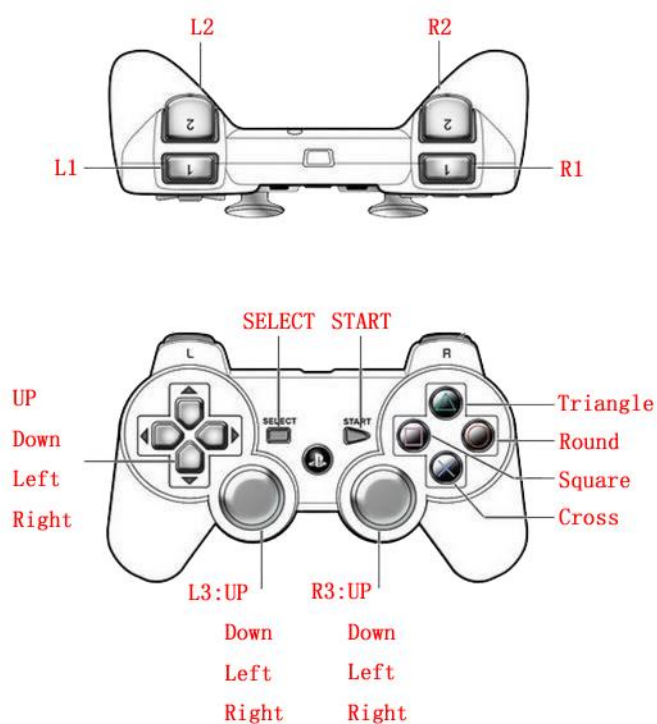
图八

使用无线遥控接收端与舵机控制器链接在一起，如图八 1-1、2-2、3-3.....

别忘记手柄也需要 2 颗电池供电。（接线正确后，打开电源接收器与遥控的 LED 灯会常亮，代表已经配对完毕。）

无线遥控具有两种遥控模式，模式 1（黄灯灭）为单个舵机控制，模式 2（黄灯亮）为动作组控制。在不同模式下，按键的功能不同，但是有些按键在两种模式下功能一样。

注意：电源启动后必须按下一次“START”启动舵机。



图九

共同按键：

SELECT: 模式切换

START: 开始生效

单个舵机控制 (32 Servo Mode) :

Square: 所有舵机移动至 2500

Cross: 所有舵机移动至 1500

Round: 所有舵机移动至 500

Triangle: 保留

硬件	32 通道	24 通道	16 通道
第一组舵机序号	1、3、5、7、9、11、13、15	1、3、5、7、9、11	1、3、5、7
第一组控制按键	L2: 第一组舵机切换, 递减次序 R2: 第一组舵机切换, 递增次序 L3-Left: 第一组舵机值递增 L3-Right: 第一组舵机值递减		
第二组舵机序号	2、4、6、8、10、12、14、16	2、4、6、8、10、12	2、4、6、8
第二组控制按键	L1: 第二组舵机切换, 递减次序 R1: 第二组舵机切换, 递增次序 R3-Left: 第二组舵机值递增 R3-Right: 第二组舵机值递减		
第三组舵机序号	17、19、21、23、25、27、29、31	13、15、17、19、21、23	9、11、13、15
第三组控制按键	Left: 第三组舵机切换, 递减次序 Right: 第三组舵机切换, 递增次序 L3-Up: 第三组舵机值递增 L3-Down: 第三组舵机值递减		
第四组舵机序号	18、20、22、24、26、28、30、32	14、16、18、20、22、24	10、12、14、16
第四组控制按键	Down: 第四组舵机切换, 递减次序 Up: 第四组舵机切换, 递增次序 R3-Up: 第四组舵机值递增 R3-Down: 第四组舵机值递减		

Wireless controller

One of the servo motor to control

Action groups to control

32 servo motor Mode

	Servo	Value
Square	All	2500
Cross	All	1500
Round	All	500
Triangle	All	1500

Apply

Close

动作组执行:

按键	L2	L1	R2	R1	Up	Left:
动作组	0	1	2	3	4	5
按键	Down	Right	L3-Up	L3-Left	L3-Down	L3-Right
动作组	6	7	8	9	10	11
按键	R3-Up	R3-Left	R3-Down	R3-Right	Square	Cross
动作组	12	13	14	15	16	17
按键	Round	Triangle				
动作组	18	19				

Wireless controller

One of the servo motor to control Action groups to control

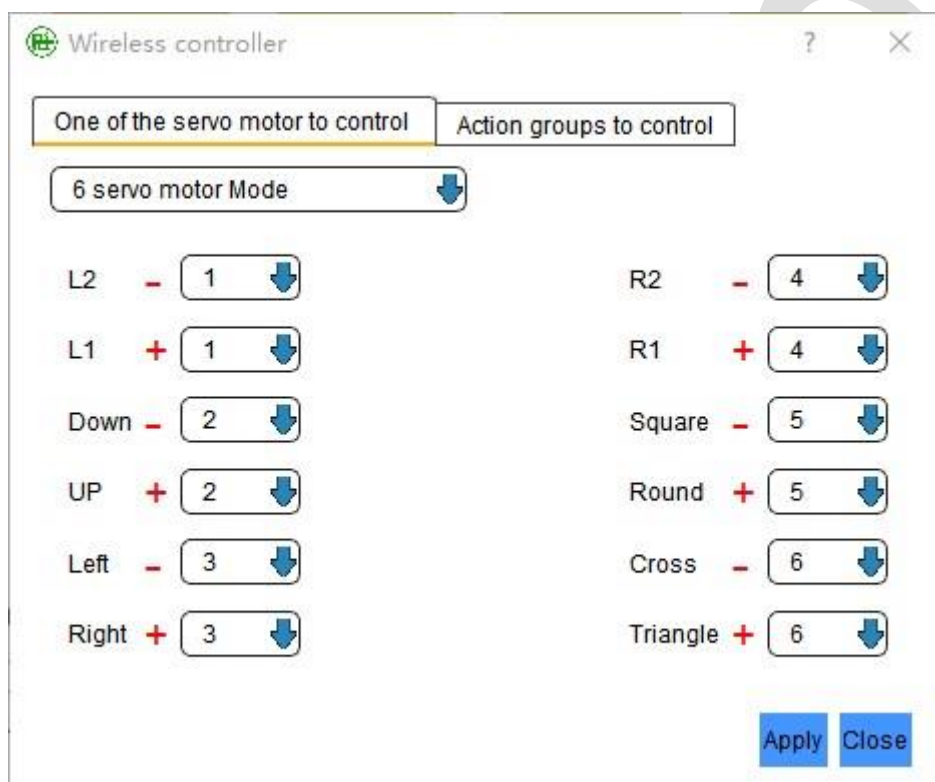
L2	G0F1	R2	G2F1
L1	G1F1	R1	G3F1
UP	G4F1	Square	G16F1
Left	G5F1	Cross	G17F1
Down	G6F1	Round	G18F1
Right	G7F1	Triangle	G19F1
L3-UP	G8F1	R3-UP	G12F1
L3-Left	G9F1	R3-Left	G13F1
L3-Down	G10F1	R3-Down	G14F1
L3-Right	G11F1	R3-Right	G15F1

Apply Close

单个舵机控制(6 Servo Mode):

每次按键指定舵机值会递增或者递减，如按下“L2”键，序号1的舵机值会减少以达到改变角度。

按键	L2	L1	R2	R1	Down	Up:
舵机序号	1-	1+	4-	4+	2-	2+
按键	Left	Right	Square	Round	Cross	Triangle
舵机序号	3-	3+	5-	5+	6-	6+

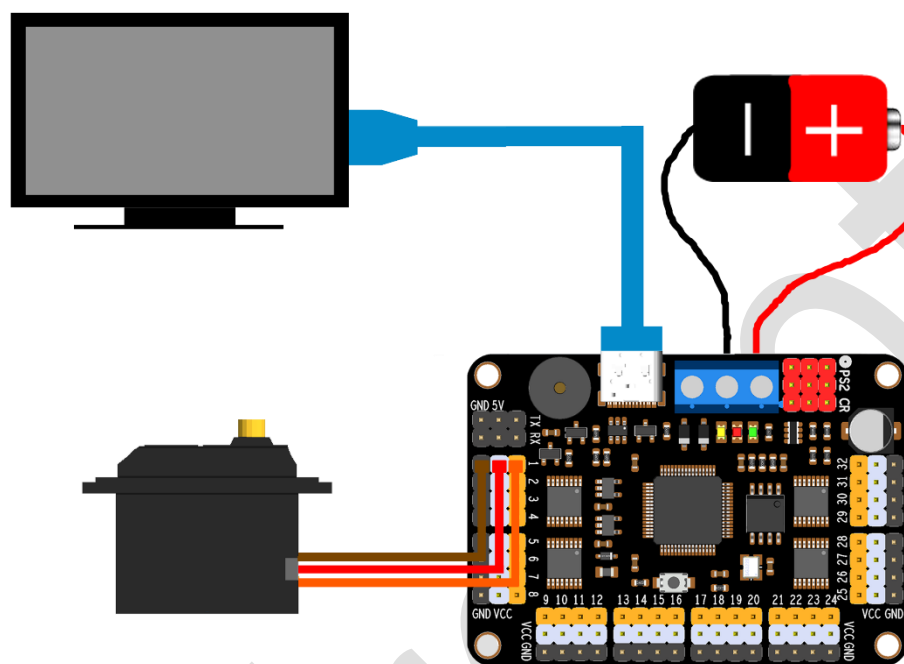


如需自定义手柄按键功能请使用软件点击“Setting” -> “Wireless controller”里编辑。

当舵机控制器接入USB启动时，无线摇杆无法使用。

整体接线例示:

一、使用电脑控制:



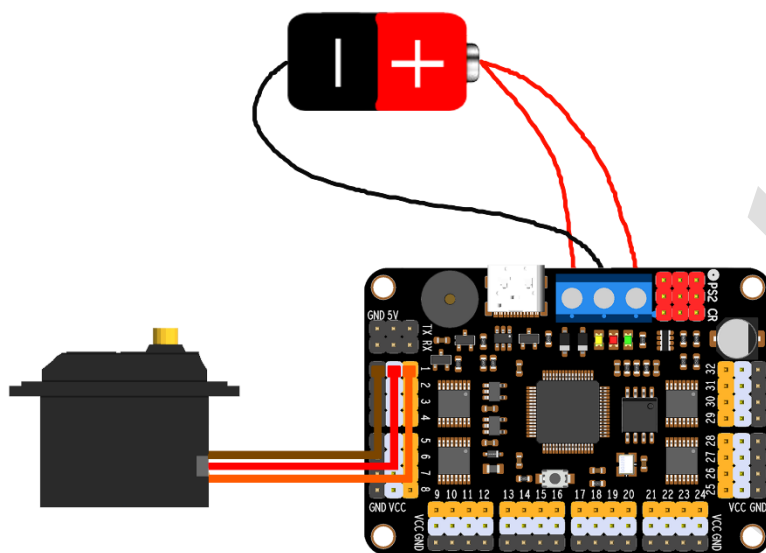
图十

先把舵机与舵机的电源接入，然后使用 USB 线把电脑与舵机控制器链接在一起。

舵机的电源接入请参考**接线方法一**（谨记 VDD 电源口不接）

温馨提示：舵机电源应根据舵机要求选择。

二、控制器自动运行：



图十一

在需要舵机自动运行之前先使用软件设置自动运行参数，详细看软件使用篇。

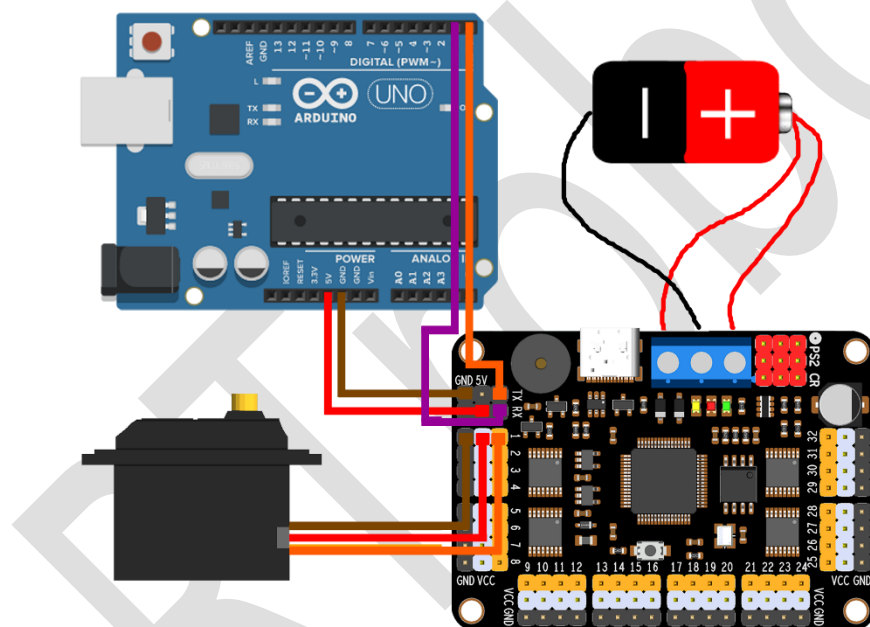
温馨提示：若需要 USB 为 CPU 供电则 VDD 口的红线不接，软件设置完自动运行参数后拔插 USB 口即会自动运行。

三、使用 MCU 控制：

控制器为 MCU 供电：

这里以 Arduino uno 为例子，其他 MCU 参考此处。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。如图十二。

注意：此时所有电源都由一个电池完成。

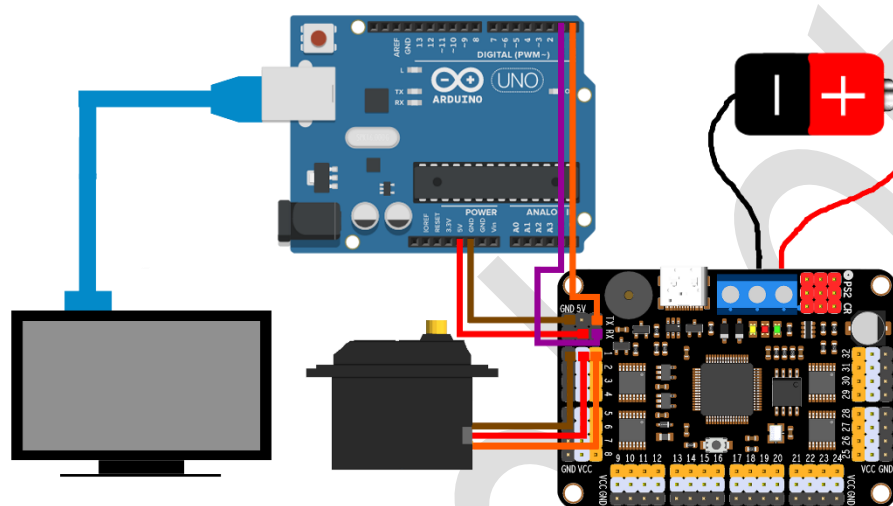


图十二

MCU 为控制器供电：

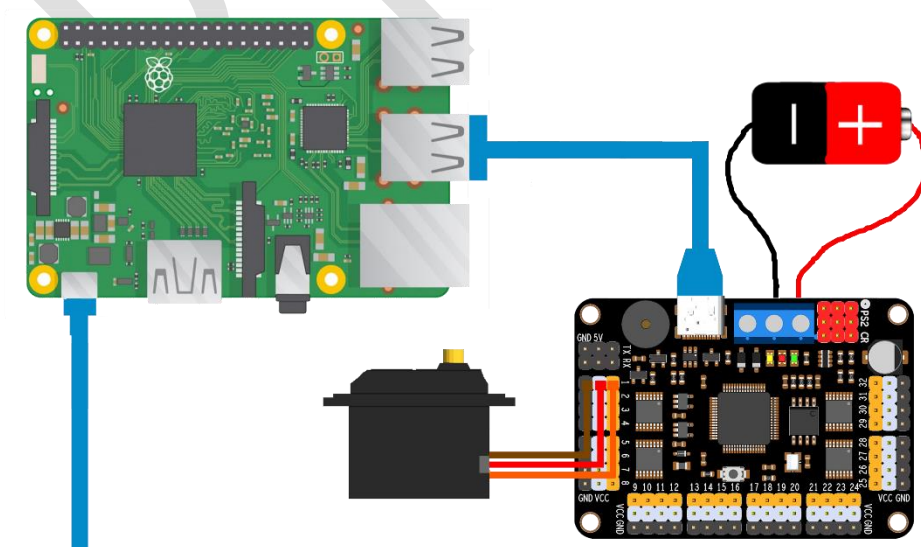
控制器 5V 接 Arduino uno 的 5V，控制器 GND 接 Arduino uno 的 GND，控制器 TX 接 Arduino uno 的 RX，控制器 RX 接 Arduino uno 的 TX。如图十三。

注意：电脑为 Arduino uno 与控制器的供电，舵机则独立供电。



图十三

Raspberry Pi 用户可以使用 USB 线将控制器连接到 Raspberry Pi 上，把它当作一台电脑。



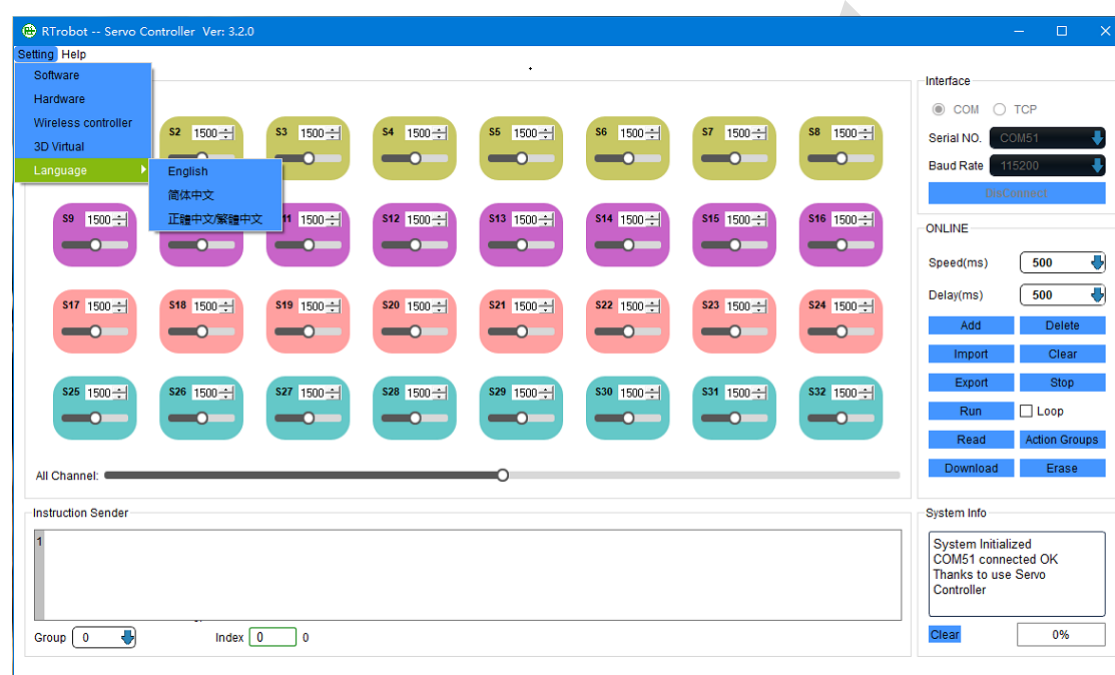
图十四

软件操作方法:

注意：需打开串口才能使用软件的全部功能

一、语言设置:

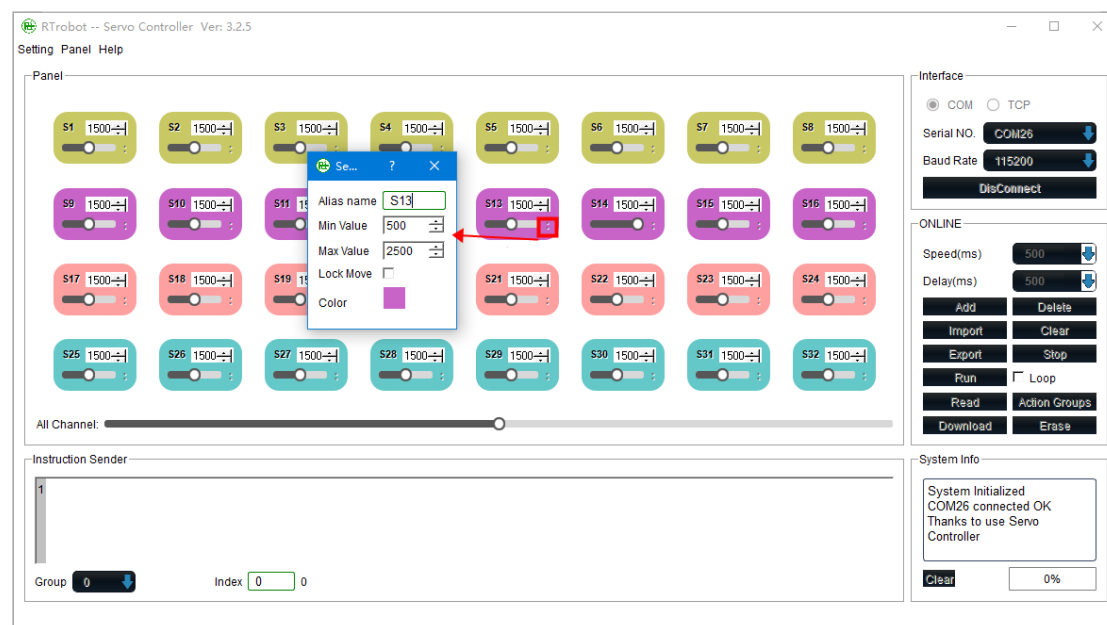
点击 “Setting” -> “Language” 选择语言（简体/繁体/English）。



图十五

二、软件设置：

点击舵机窗口的 “：”，可以单独对每个舵机命名与设置最大最小值、颜色和锁定位置。



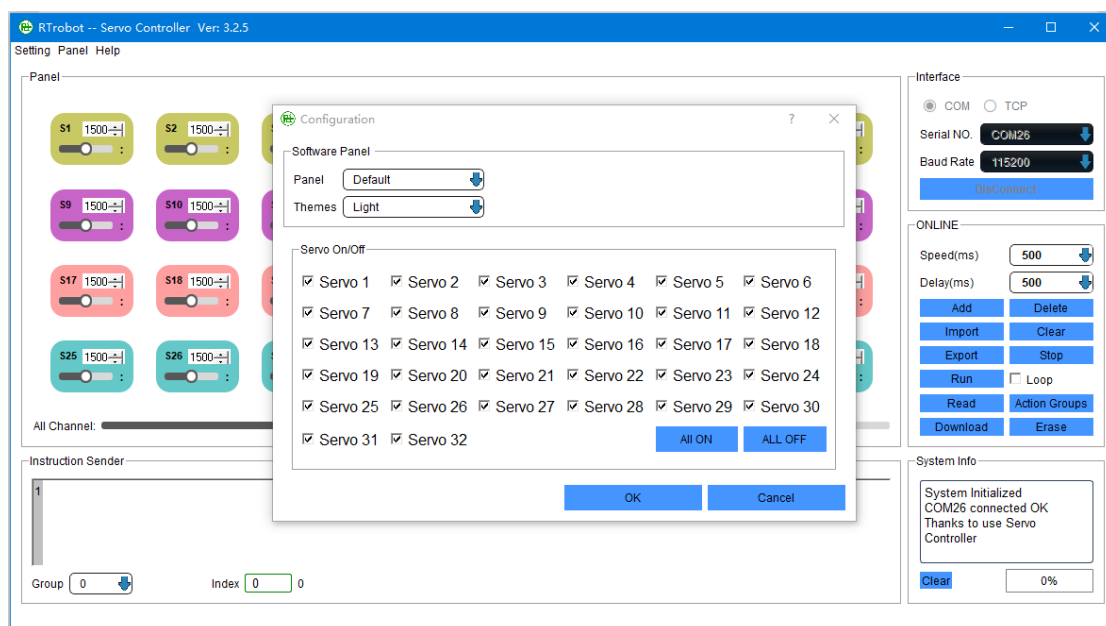
图十六

点击 “Setting” -> “Software” 可对软件进行设置，如图十六。

Software Panel：设置软件控制面板。

Servo On/Off：显示需要控制的舵机序号。

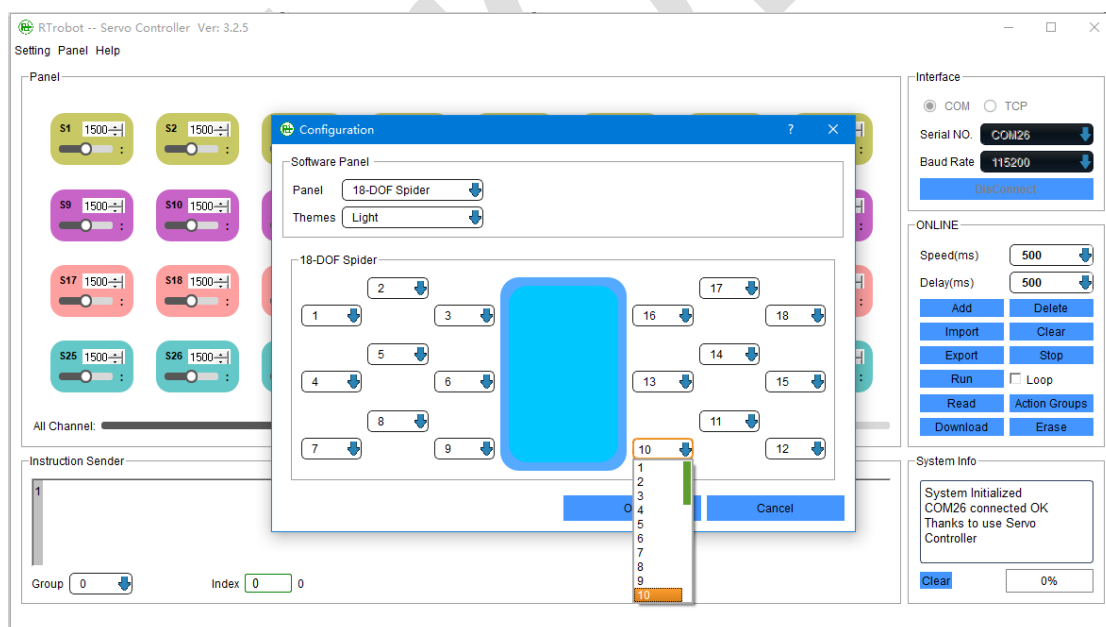
软件设置完成后点击 “OK” 会自动重新启动软件。



图十七

Software Panel 选择界面后可指定每个位置的舵机序号，如图十八。

注意：如果存在重复序号的舵机，则无法保存。



图十八

三、控制器设置：

点击 “Setting” -> “Hardware” 可对控制器进行设置，如图十九。

Servo initial value: 设置每个舵机上电启动的初始值。

Servo Deviation Value: 设置每个舵机的偏差值 (有效值: **-99~99**)，如图二十。

Uart Baud Rate: 设置图五中④位置的串口波特率。

Buzzer: 低压警报开关。

Start Automatic run: 开机自动运行动作组开关。

Automatic run group: 设置开机自动运行的动作组序号，当自动运行选择为仅动作组模式时，此选项无效。

Automatic run times: 动作组开机运行次数，当自动运行选择为仅动作组模式时，此选项无效。

注意：设置完毕后记住点击 Apply，等待设置完毕。每次设置后控制器需要重新启动才会生效。

The screenshot shows the 'Hardware' settings window with the 'Servo initial value' tab selected. It contains a 3x10 grid of input fields for Servo 1 through Servo 32, and an 'All Servo' field. All fields are set to 1500. Below the grid are settings for 'UART Baud Rate' (9600), 'Buzzer' (Open), 'Start automatic run' (No), 'automatic run group' (0), and 'automatic run times(max:254; loop:0;)' (0). 'Apply' and 'Close' buttons are at the bottom.

图十九

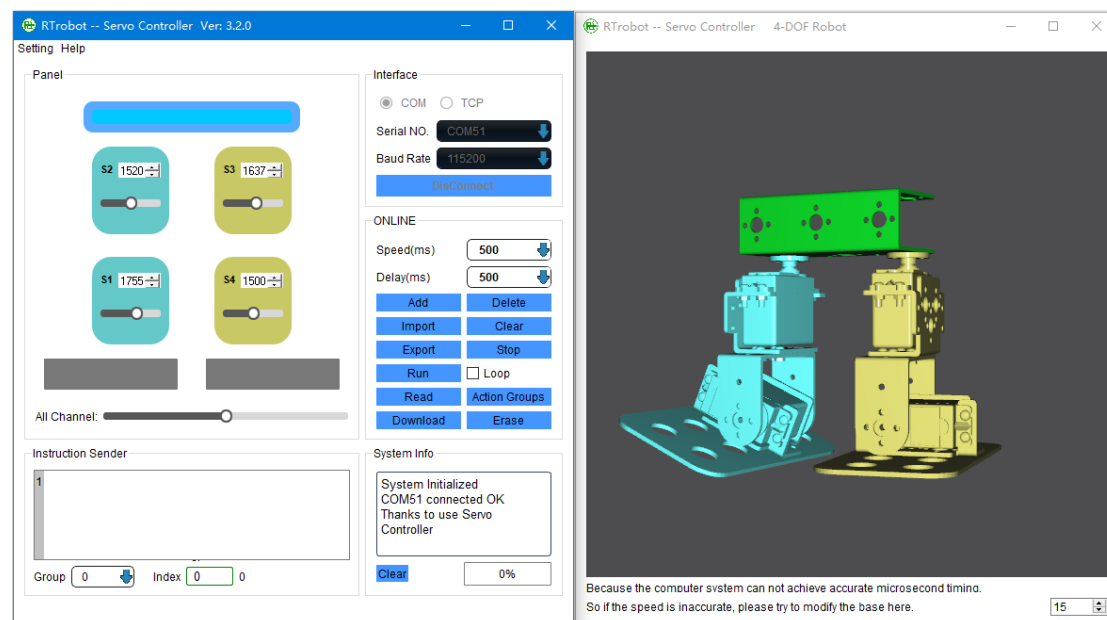
The screenshot shows the 'Hardware' settings window with the 'Servo deviation value' tab selected. It contains a 3x10 grid of input fields for Servo 1 through Servo 32, and an 'All Servo' field. All fields are set to 0. Below the grid are settings for 'UART Baud Rate' (9600), 'Buzzer' (Open), 'Start automatic run' (No), 'automatic run group' (0), and 'automatic run times(max:254; loop:0;)' (0). 'Apply' and 'Close' buttons are at the bottom.

图二十

四、3D 展示效果：

点击 “Setting” -> “3D Virtual” 可展示 3D 效果界面

注意：需要先在 “Software” 设置页选择好控制界面后打开串口才能展示 3D 效果界面



图二十一

五、软件控制：

1. 选择一种合适自己的接线方式，并使用 USB 线与电脑连接好。
2. 安装控制器的驱动 (Servos Controller Drive.exe) (Windows10 以上无需安装驱动)。

注意：如提示无数字签名无法安装，则把系统禁用驱动数字签名方式启动电脑系统再次运行安装。

3. 打开软件 “ServoController.exe”。
4. 选择好串口，并且打开串口。如果使用 WIFI 模式，选择 TCP，填写好 WIFI 模块 TCP 服务器设定的 TCP 地址与端口。

注意：只有使用 USB 线连接电脑时才能使用全部功能。

① 单个舵机操作:

如图二十二所示，拖动或者是填写都可以改变舵机角度值。



图二十二

② 多个舵机操作:

如图二十三所示，先在指令信息框下面“Group”选择框中选择好要编辑的动作组序号后再设置每个舵机的第一行运行值，并且设置舵机的运行速度“Speed”，与执行完毕后的延迟等待时间“Delay”。点击“Add”添加，然后再设置每个舵机的第二行运行值，点击“Add”添加。所有预设动作设置完毕后，点击“Run”进行测试。如果“Loop”勾选上则会无限循环运行。

舵机运行速度“Speed”：在指定时间内完成指令（不可超过舵机物理最大速度）。

舵机运行完毕等待时间“Delay”：执行完成当前指令后延迟指定的时间，再执行下一条指令。



图二十三

③ 动作指令保存

点击“Export”把动作指令保存到文本中，以便下次导入使用。

注意：这里保存为全部动作组的指令。

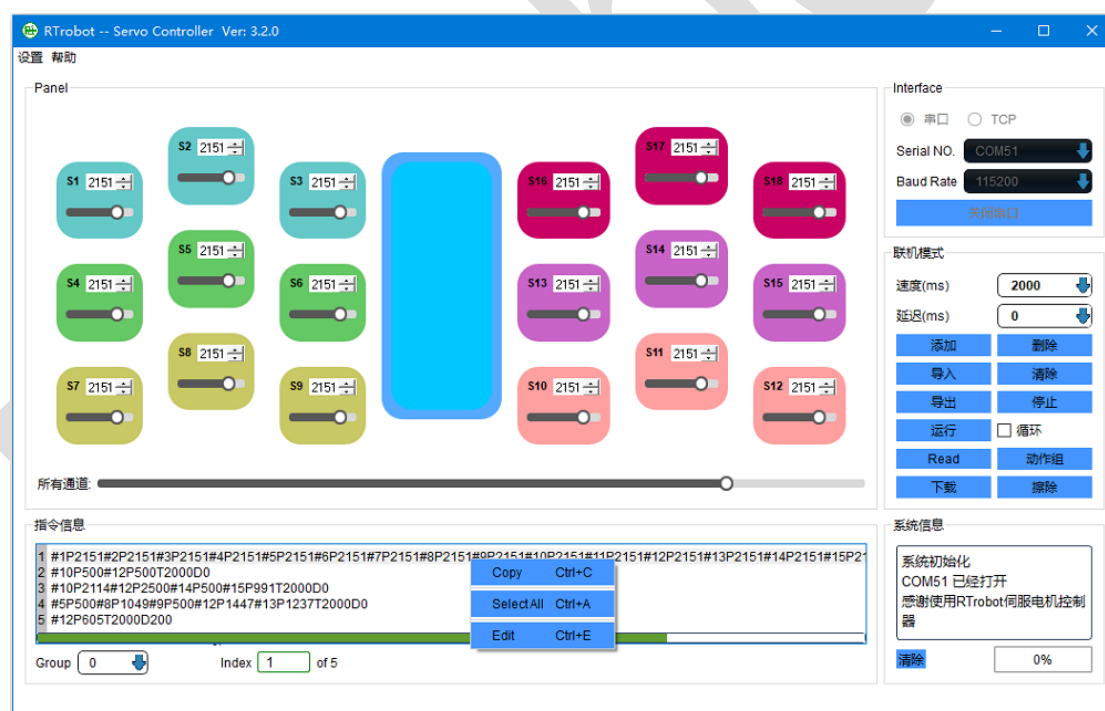
④ 使用文件导入操作：

点击“Import”把以前保存的动作指令导入到软件中。

注意：这里导入为全部动作组的指令。

⑤ 动作指令编辑：

在指令信息框里点击需要编辑的那行指令，右键选择“Edit”或者快捷键“Ctrl+E”进行编辑



图二十四

⑥ 脱机运行：

待所有指令编辑好后点击“Download”下载所有动作组的指令，

在“Setting” -> “Hardware” 界面中开启控制器自动运行开关与运行的动作组序号。

⑦ 读取指令：

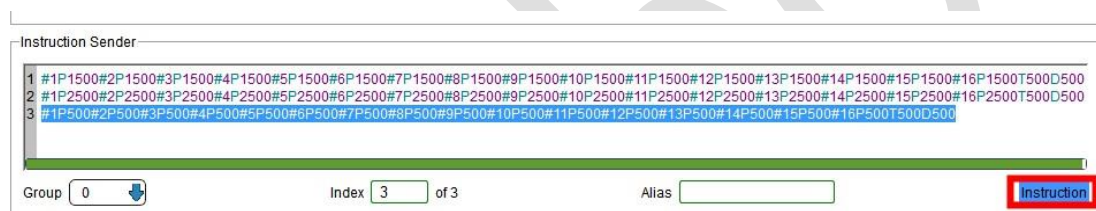
点击“Read”可读取已经下载至控制器的所有指令。

⑧ 擦除所有动作组：

点击“Erase”可把所有已经下载至控制器的指令擦除，擦除时间约 30 秒。

⑨ 手动编辑指令

点击“Instruction”可在弹出的对话框中手动输入或编辑指令。



图二十五

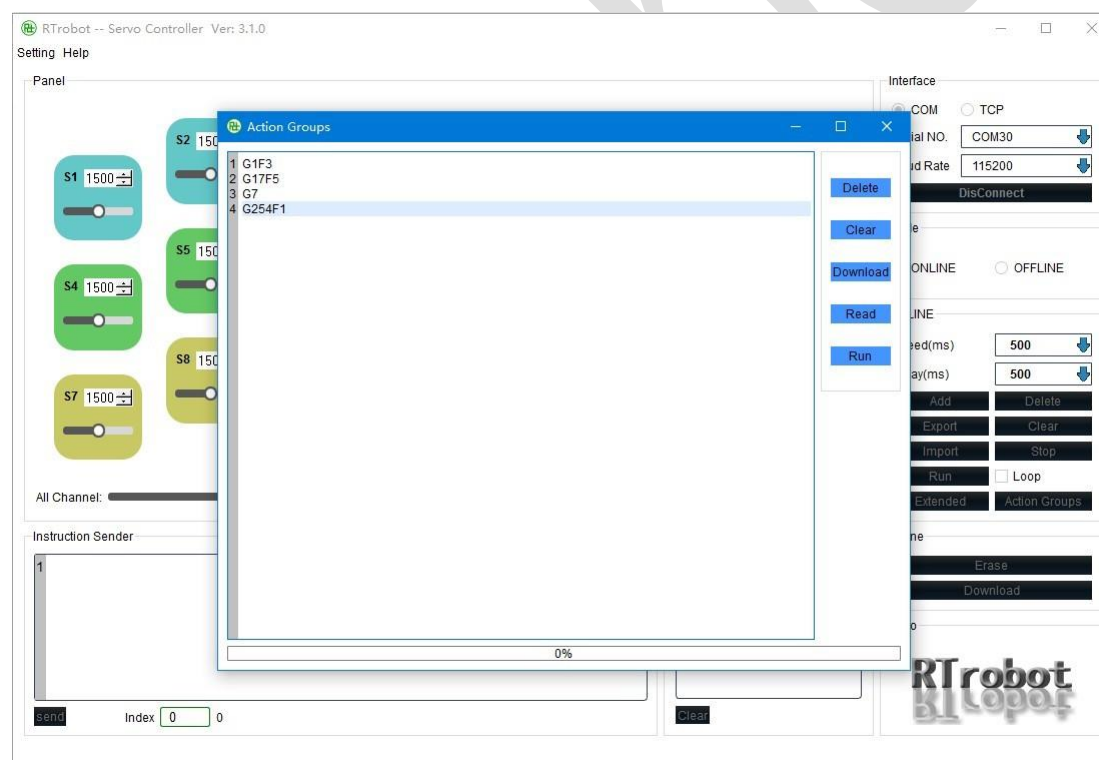
六、动作组编辑：

点击 “Action group” 打开动作组运行编辑窗口，如图二十六；此处只能编辑动作组指令，如：

G1F3
G17F5
.....

G1 表示动作组 1，F3 表示运行 3 次。

编辑完成后可以点击 “Run” 进行测试，测试无误后便可下载保存至控制器中，下次打开软件时可以使用 “Read” 功能把之前下载的动作组指令读取出来；若需自动运行，请在硬件设置中把自动运行开关改为仅动作组。



图二十六

七、MPU6500(仅 24 通道控制器):

点击 “Setting” -> “MPU6500” 打开 MPU6500 设置界面，如图二十七；点击

“Disable” 键改变状态为 “Enable” 使 MPU6500 开启。Pitch、Roll、Yaw 为

MPU6500 的 XYZ 值，填写好允许偏差值、超出范围后需要运行的动作组与运行次数后，

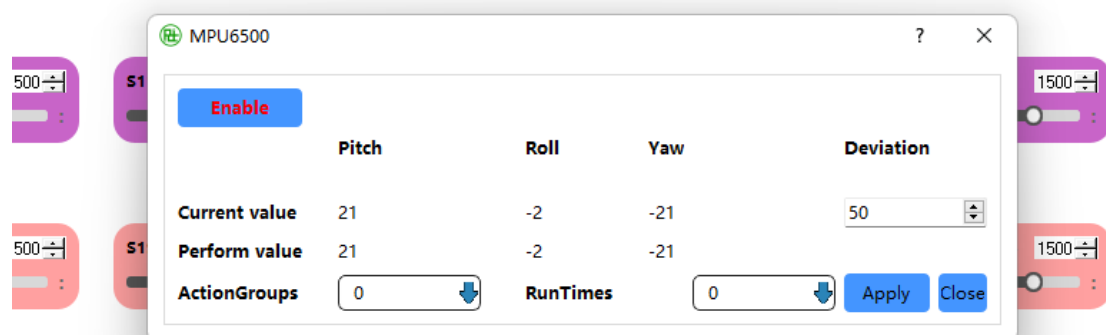
把控制器摆成触发运行状态并按下应用键使其生效后重启控制器。当控制器倾斜方向达到

设置的值时，即会触发并且运行之前设定的动作组与运行次数。（若使用 USB 线链接电脑

并使用上位机软件调试的时候，MPU6500 不会触发。）

若运行次数设置为 “0” ，则不会触发，只会通过串口反馈 MPU6500 的值。

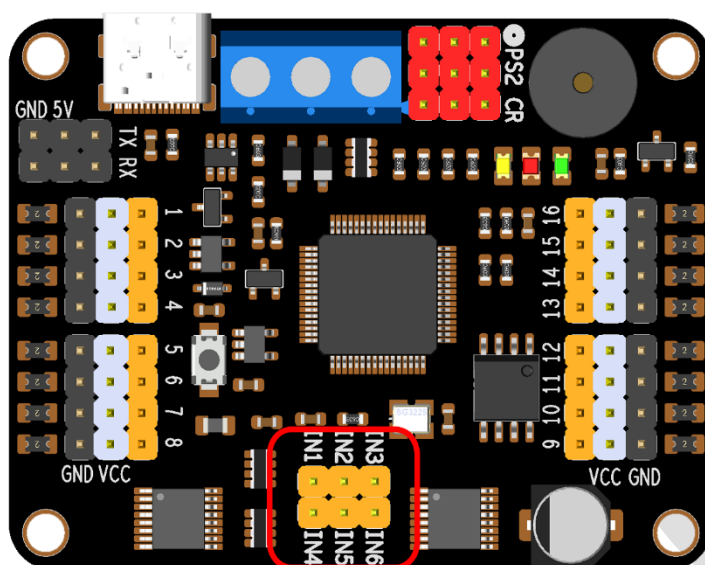
21	,	-2	,	-21	\r	\n
0X32 0X31	0X2C	0X2D 0X32	0X2C	0X2D 0X32 0X31	0X0D	0X0A
Pitch	,	Roll	,	Yaw	\r	\n



图二十七

注意：执行动作组前串口会首先反馈 “TRIGGER”，执行完成指定动作组后反馈 “OK”。

八、6 通道数字电平传感器接口(仅 16 通道控制器):



图二十八

图二十八中红圈处的 6 个数字电平传感器接口，每个传感器可以独立控制 6 个动作组或者是保持指定舵机的当前位置（只支持 3.3/5V 数字电平传感器）。

若使用 USB 线链接电脑并使用上位机软件调试的时候，外部传感器不会触发。

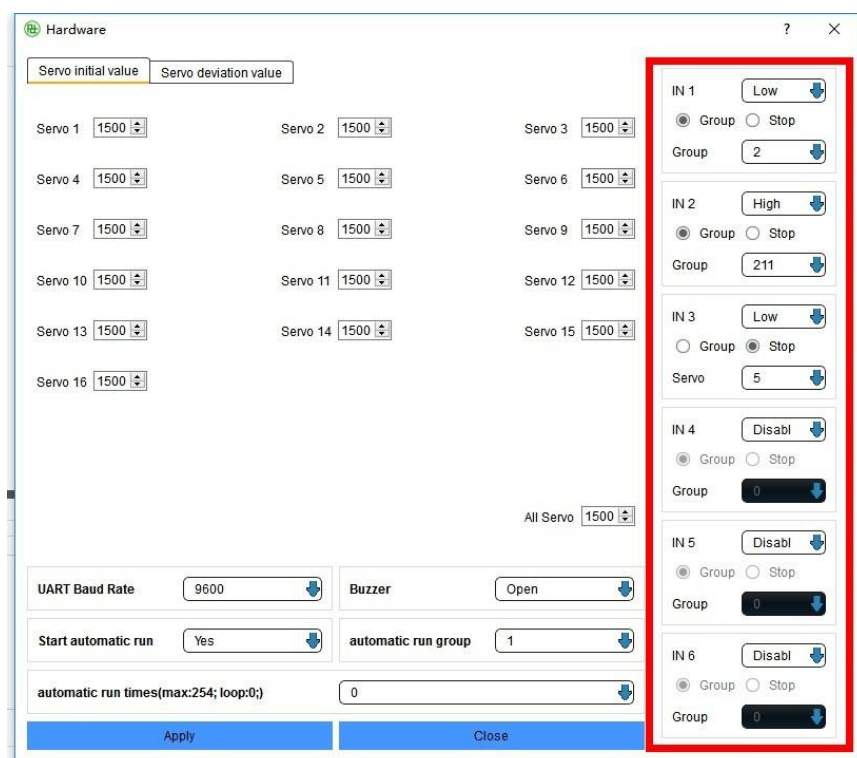
注意：每个传感器的 GND 都需要与控制器的 GND 接在一起。

当多个 IN 同时触发时，则序号小的 IN 有效；

$IN1 > IN2 > IN3 > IN4 > IN5 > IN6$

例 1：IN2 与 IN3 同时触发，只会执行 IN2 指定的动作组，若 IN2 释放，IN3 保持触发，则执行 IN3 指定的动作组。

例 2：IN1 触发后执行动作组，IN6 为保持指舵机当前的位置；当同时触发时，两者互不影响，IN1 与 IN6 同时有效。



图二十九

INx 三个选项：

- Disable：关闭（触发无效）
- High：高电平触发
- Low：低电平触发

Group：触发后执行的动作组。

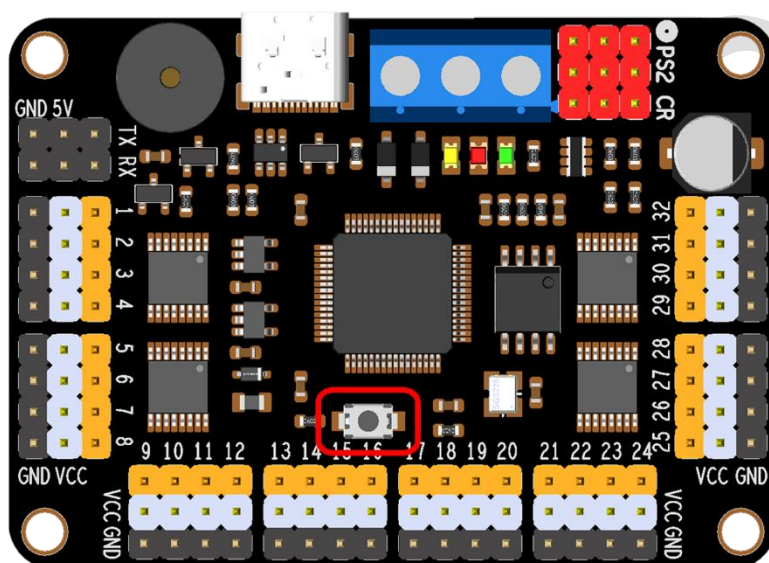
Stop：触发后停止舵机并保持当前位置。

注意：执行动作组前串口会首先反馈“TRIGGER”，执行完成指定动作组后反馈“OK”。

注意：AC 电平传感器无效。

九、升级固件方法

1. 在 "<http://www.rtrobot.org/software/>" 网站下载最新的上位机软件
2. 打开最新上位机软件
3. 按住舵机控制器上的按键不放的状态下链接 USB 数据线后松开按键

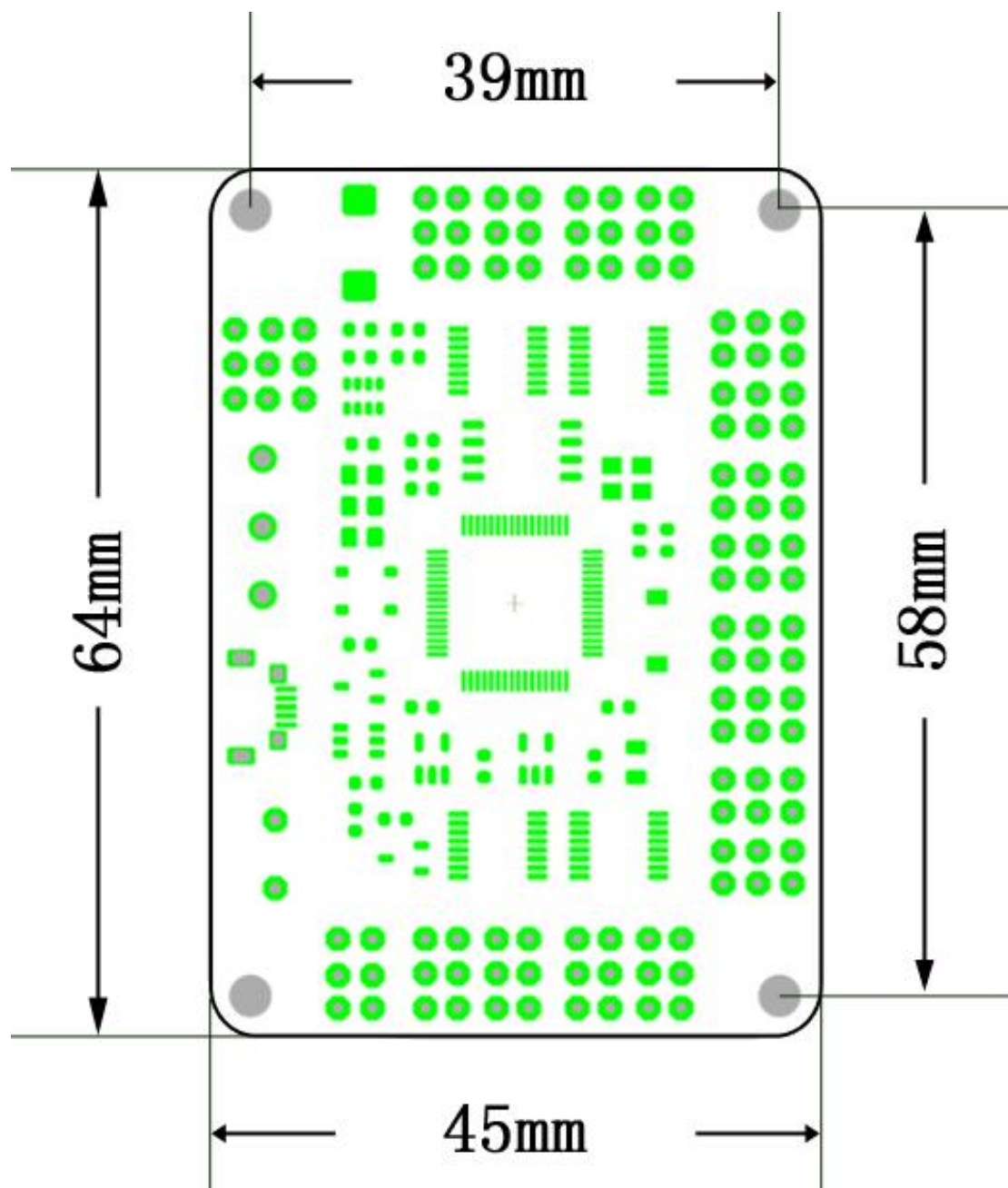


图三十

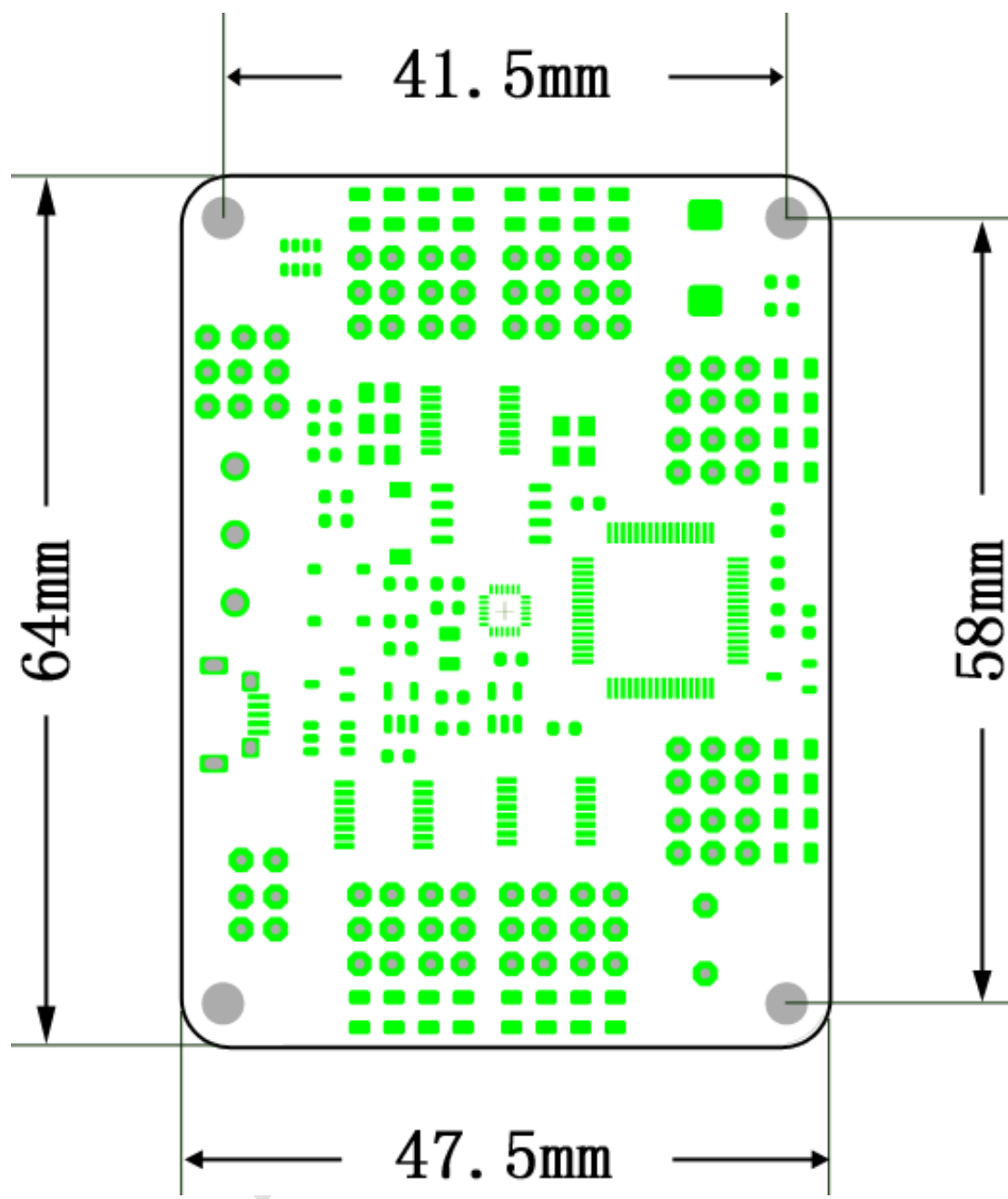
4. 打开最新上位机软件的串口，这时固件开始升级，升级完后会提示 "Update Success, Restart the controller, Please. "
5. 重启舵机控制器。
6. 如已是最新固件的话会提示 "ERROR Don't need Update!"

尺寸图:

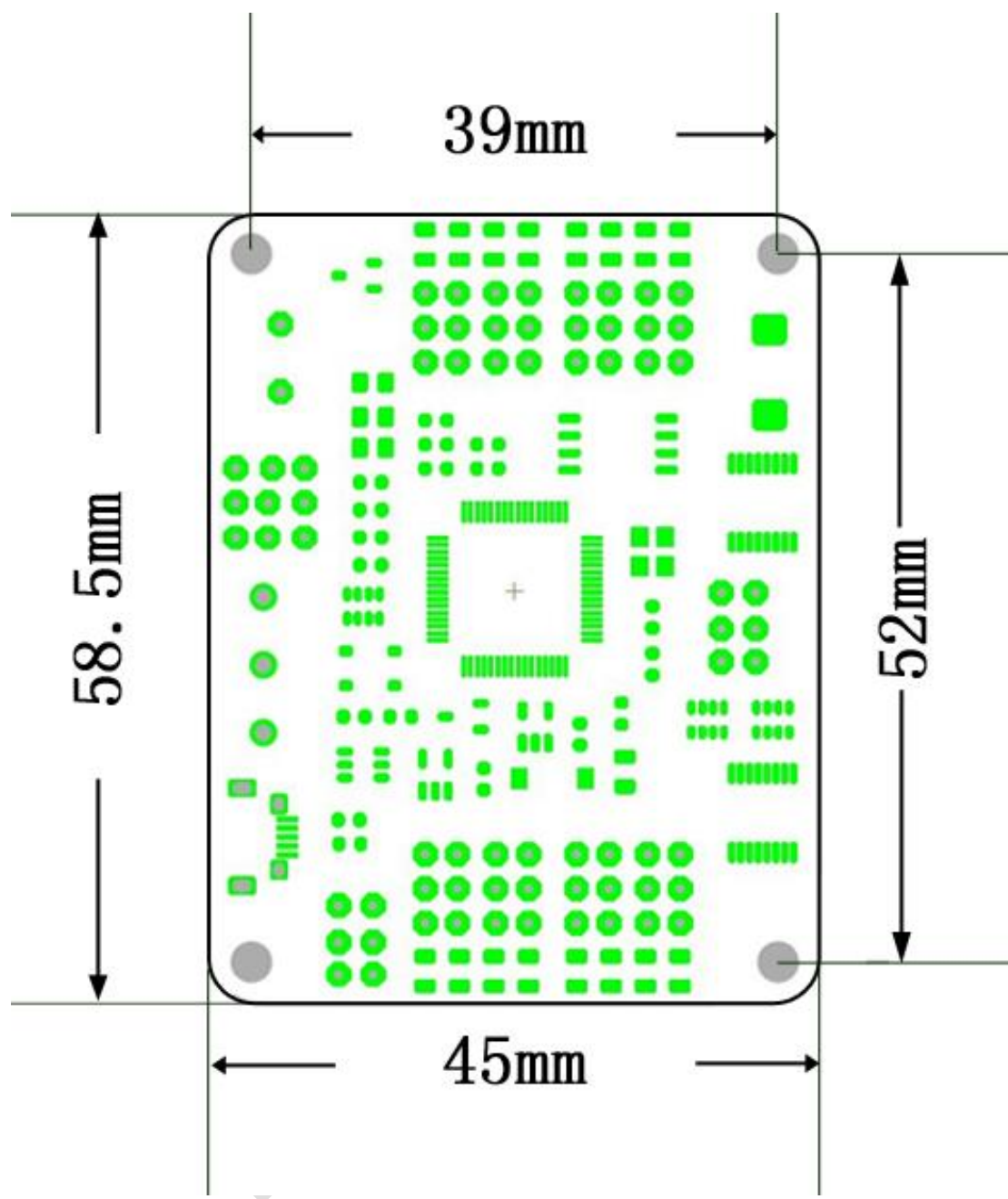
32:



24:



16:



十、关于:

感谢使用 RTrobot 出品的舵机控制器，在使用中对控制器有任何疑问需要咨询可

EMAIL:admin@rtrobot.org。