

Can舵机控制板使用说明 V0.8.5

前言

某宝无意淘到好货：60K机器人数字舵机，发现无法驱动。

基于网络整理了一些资料。

仅供学习交流使用，禁止商业用途！

串口预留：支持WIFI，蓝牙，红外、433等外部控制。

支持树莓派，UNO，Stm32、51等任意单片机串口控制。

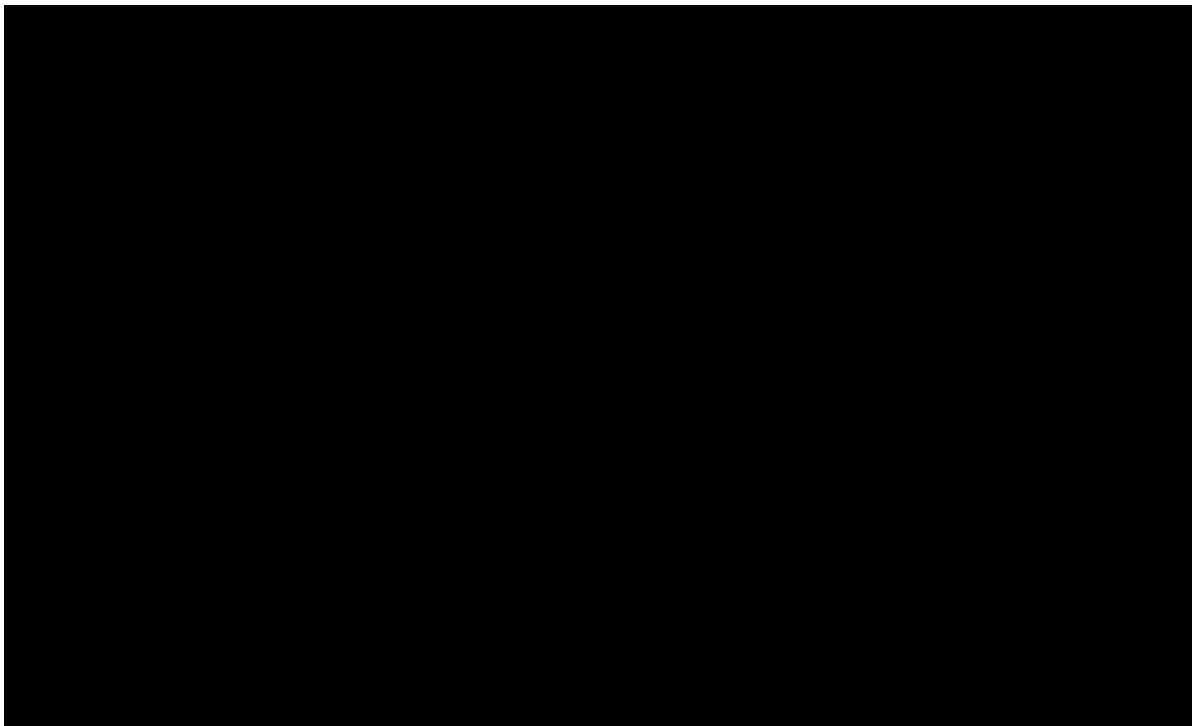
控制板就是管理多个舵机协调工作的系统，给个指令就可以控制，非常的友善。

驱动板说明

电源建议：

电源：使用24V3A动力电源、或者6S电池组(6节18650电池)。

接线示意图：



控制指令

为了简单明了，这里挑选几个常用命令：

[illegible]

SD卡音乐示例				
名称		修改日期	类型	
01		2020/10/9 21:15	文件夹	
02	SD卡目录结构	2020/10/9 21:42	文件夹	
03	01,02,03.....是音频目录	2020/10/9 21:42	文件夹	
MP3	Mp3.....是系统音频目录	2020/10/9 20:36	文件夹	

Arduino编程示例

编程非常方便。

通过编程来管理控制板是非常的方便的，发送一条指令就可以控制了。

列一：UNO编程

```
//使用软串口
//V-V(5V) G-G T-R R-T
#include <SoftwareSerial.h>
SoftwareSerial softser = SoftwareSerial(11, 12); //软串口

byte Cmd_1[7] = {0xA9, 0x9A, 0x03, 0x41, 0x02, 0x46, 0xED}; //播放：动作组2
byte Cmd_2[6] = {0xA9, 0x9A, 0x02, 0x4F, 0x51, 0xED}; //停止播放：动作组、动作

void setup() {
    softser.begin(115200);
    delay(10);
}

void loop() {
    softser.write(Cmd_1, 7);
    delay(1000*13);
    softser.write(Cmd_2, 6);
    delay(3);
}
```

列二：Mage2560

```
//使用串口3
//V-V(5V) G-G T-R R-T
#include <Arduino.h>

byte Cmd_1[7] = {0xA9, 0x9A, 0x03, 0x41, 0x02, 0x46, 0xED}; //播放：动作组2
byte Cmd_2[6] = {0xA9, 0x9A, 0x02, 0x4F, 0x51, 0xED}; //停止播放：动作组、动作
byte Cmd_3[13] = {0xA9, 0x9A, 0x09, 0x88, 0x06, 0x00, 0x01, 0x5A, 0x00, 0xF4, 0x01, 0xE7, 0xED}; //2号舵机90度
byte Cmd_4[13] = {0xA9, 0x9A, 0x09, 0x88, 0x06, 0x00, 0x01, 0xB4, 0x00, 0xF4, 0x01, 0x41, 0xED}; //2号舵机180度
```

```

void setup() {
    Serial3.begin(115200);
    delay(6000);
}
void loop() {
    for (int i = 0; i < 8; i++)
    {
        delay(8000);
        Serial3.write(Cmd_3, 13);
        delay(8000);
        Serial3.write(Cmd_4, 13);
        delay(8000);
    }
}

```

列三：PS2手柄

```

//PS2手柄 -->接在UNO板上
//驱动板 -->接在UNO板上
#include <Arduino.h>
#include <SoftwareSerial.h>
//开软串口
SoftwareSerial softser = SoftwareSerial(11, 12); //UNO板上, 接PS2手柄io
SoftwareSerial ubx_softSerial(7, 8); //-RX,TX //UNO板上, 接驱动板io
#define my_serial ubx_softSerial
int temp_char;

void setup(){
    Serial.begin(115200); //UNO硬串口, 空置未使用
    my_serial.begin(115200); //驱动板用
    softser.begin(115200); //PS2手柄用
}

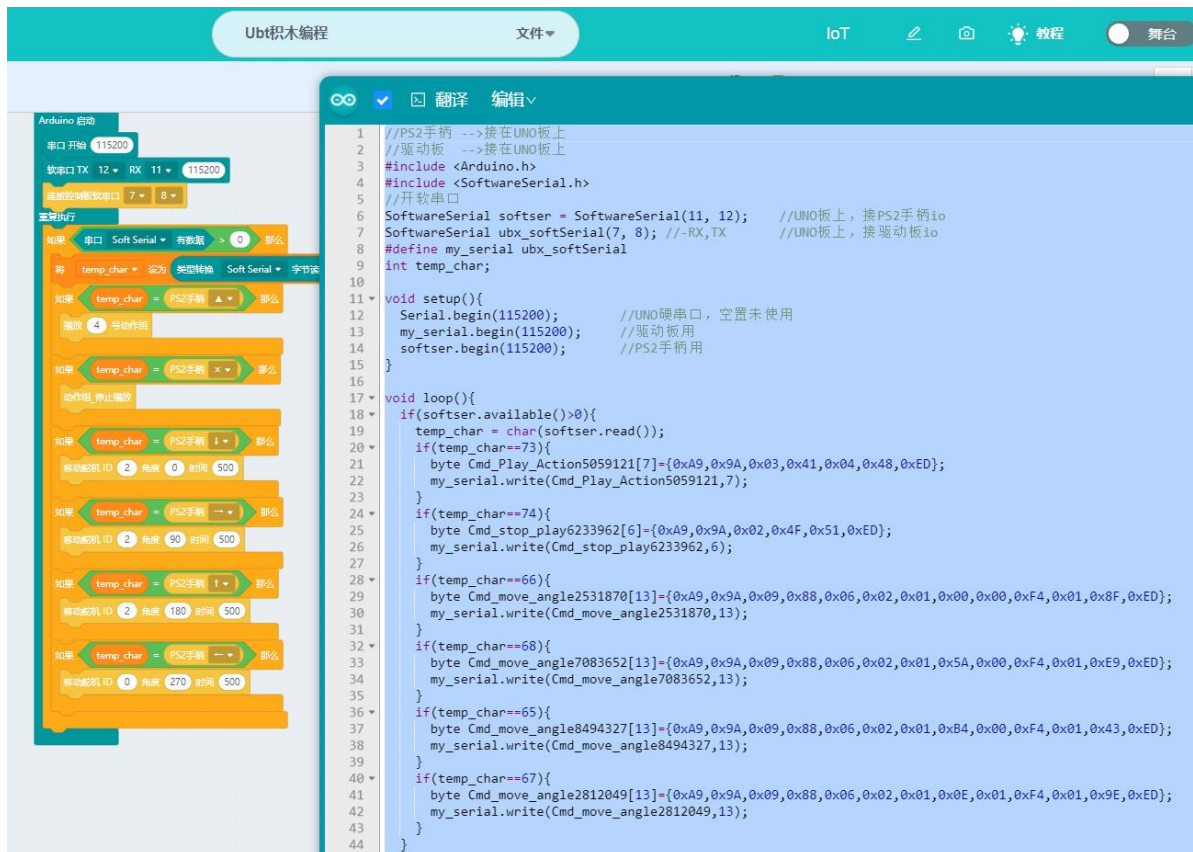
void loop(){
    if(softser.available()>0){
        temp_char = char(softser.read());
        if(temp_char==73){
            byte Cmd_Play_Action5059121[7]={0xA9,0x9A,0x03,0x41,0x04,0x48,0xED};
            my_serial.write(Cmd_Play_Action5059121,7);
        }
        if(temp_char==74){
            byte Cmd_stop_play6233962[6]={0xA9,0x9A,0x02,0x4F,0x51,0xED};
            my_serial.write(Cmd_stop_play6233962,6);
        }
        if(temp_char==66){
            byte Cmd_move_angle2531870[13]=
{0xA9,0x9A,0x09,0x88,0x06,0x02,0x01,0x00,0x00,0xF4,0x01,0x8F,0xED};
            my_serial.write(Cmd_move_angle2531870,13);
        }
        if(temp_char==68){
            byte Cmd_move_angle7083652[13]=
{0xA9,0x9A,0x09,0x88,0x06,0x02,0x01,0x5A,0x00,0xF4,0x01,0xE9,0xED};
            my_serial.write(Cmd_move_angle7083652,13);
        }
        if(temp_char==65){
            byte Cmd_move_angle8494327[13]=
{0xA9,0x9A,0x09,0x88,0x06,0x02,0x01,0xB4,0x00,0xF4,0x01,0x43,0xED};

```

```

        my_serial.write(Cmd_move_angle8494327,13);
    }
    if(temp_char==67){
        byte Cmd_move_angle2812049[13]={
0xA9,0x9A,0x09,0x88,0x06,0x02,0x01,0x0E,0x01,0xF4,0x01,0x9E,0xED};
        my_serial.write(Cmd_move_angle2812049,13);
    }
}
}
}

```



列四：树莓派编程

#树莓派和控制通讯，实例

#usb线一端接树莓派，另一端接控制板【或者接在板子的黄色排针上】

#这里树莓派使用的是：/dev/ttyUSB0【请根据实际情况设置】

```

import serial
import time
import binascii
s = serial.Serial('/dev/ttyUSB0', 115200)

```

```

angle_90= [0xA9,0x9A,0x09,0x88,0x06,0x00,0x01,0x5A,0x00,0xE8,0x03,0xDD,0xED] #90度
angle_180=[0xA9,0x9A,0x09,0x88,0x06,0x00,0x01,0xB4,0x00,0xE8,0x03,0x37,0xED] #180度
angle_270=[0xA9,0x9A,0x09,0x88,0x06,0x00,0x01,0x0E,0x01,0xE8,0x03,0x92,0xED] #270度
angle_360=[0xA9,0x9A,0x09,0x88,0x06,0x00,0x01,0x67,0x01,0xE8,0x03,0x09,0xED] #360度
play_id_hex = [0xA9,0x9A,0x03,0x41,0x03,0x47,0xED] # 播放动作组3

```

```

s.write(b'\xA9\x9A\x02\xF4\x51\xED') #先停止,其他动作

```

```

senddata=angle_90
s.write(senddata)
time.sleep(2)

senddata=angle_180
s.write(senddata)
time.sleep(2)

senddata=angle_270
s.write(senddata)
time.sleep(2)

senddata=angle_360
s.write(senddata)
time.sleep(2)

#senddata=play_id_hex      #
#s.write(senddata)
#time.sleep(2)

s.close()

```

V2 192.168.1.8 (raspberrypi) - VNC Viewer

Thonny - /home/pi/Desktop/02.py @ 17:18

File Edit View Run Tools Help

02.py

```

#树莓派和控制通讯，实例
#usb线一端接树莓派，另一端接控制板[或者接在板子的黄色排针上]
#这里树莓派使用的是:/dev/ttyUSB0【请根据实际情况设置】
import serial
import time
import binascii
s = serial.Serial('/dev/ttyUSB0', 115200)

angle_90= [0xA9,0x9A,0x09,0x88,0x06,0x00,0x01,0x5A,0x00,0xE8,0x03,0xDD,0xED] #90度
angle_180=[0xA9,0x9A,0x09,0x88,0x06,0x00,0x01,0xB4,0x00,0xE8,0x03,0x37,0xED] #180度
angle_270=[0xA9,0x9A,0x09,0x88,0x06,0x00,0x01,0x0E,0x01,0xE8,0x03,0x92,0xED] #270
angle_360=[0xA9,0x9A,0x09,0x88,0x06,0x00,0x01,0x67,0x01,0xE8,0x03,0x09,0xED] #360度
play_id_hex = [0xA9,0x9A,0x03,0x41,0x03,0x47,0xED] # 播放动作组3

s.write(b'\xA9\x9A\x02\x4F\x51\xED') #先停止,其他动作

senddata=angle_90
s.write(senddata)
time.sleep(2)

```

Shell

```

>>> %Run 02.py
>>> |

```

Thonny - /home/pi/...

16% 22:57

示教动作组

示教功能分2种情况：

第一种，在电脑上用上位机操作。配套资料包路面有操作视频资料里面有；

第二种，离线示教。配合红外遥控器操作非常的方便。

这里我们来讨论第二种方式的实现。

上位机软件右边依次点击：**主控板-->特殊动作-->事件处理系统**：

默认情况下：示教动作组设置为6#【可以自己修改】，编辑动作的过程中OLED屏幕上会显示动作组Id号，第几条记录。

红外遥控器设置如下：

*：所有舵机解锁。【舵机解锁后才能拖动关节】

OK：示教动作添加1条Pos记录

4：删除动作组6#，【动作不满意，用次命令删除动作组重来】

6：播放动作组6#，【播放刚才编辑好的动作组】

如果打开“事件处理系统”里面没有加载积木设置，也可以在“事件处理系统”窗口的左下角点击**“载入档案”**，加载文件选择：“示教动作组_调试”来手动加载。

示教流程（默认绑定到6#动作组）：

1.按“*”键，解锁所有舵机。

2.用手搬动舵机关节到指定的位置。

3.按“OK”键，添加一帧动作数据。

.....循环第2，3步。直到动作组编辑完成。

4.按“6”键，查看刚才拖动示教的6#动作组。

如果动作不满意，按“4”键，删除对应的6#动作组，重新拖动示教即可。

编辑好了，如想更细微的调整可在上位机软件里面进行操作。



升级固件

如果正常使用过程中，又不想体验新功能的情况下，可以选择不升级固件。有时候新的固件程序还需要更新对应的上位机软件。（是否更新请谨慎考虑。）

固件更新方式：

从上位机软件左上角菜单选择：**文件**----->**机器人固件烧录**----->

烧录固件的窗口出来后：**窗口左边选择控制板对应的串口**----->**在线升级**。

