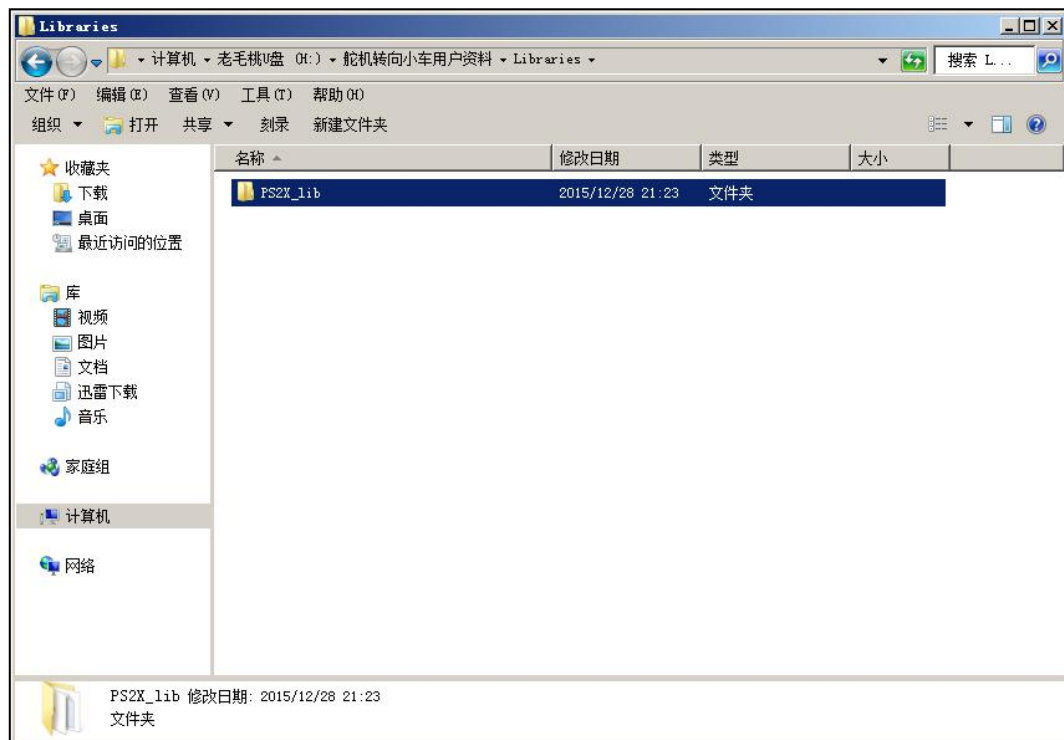
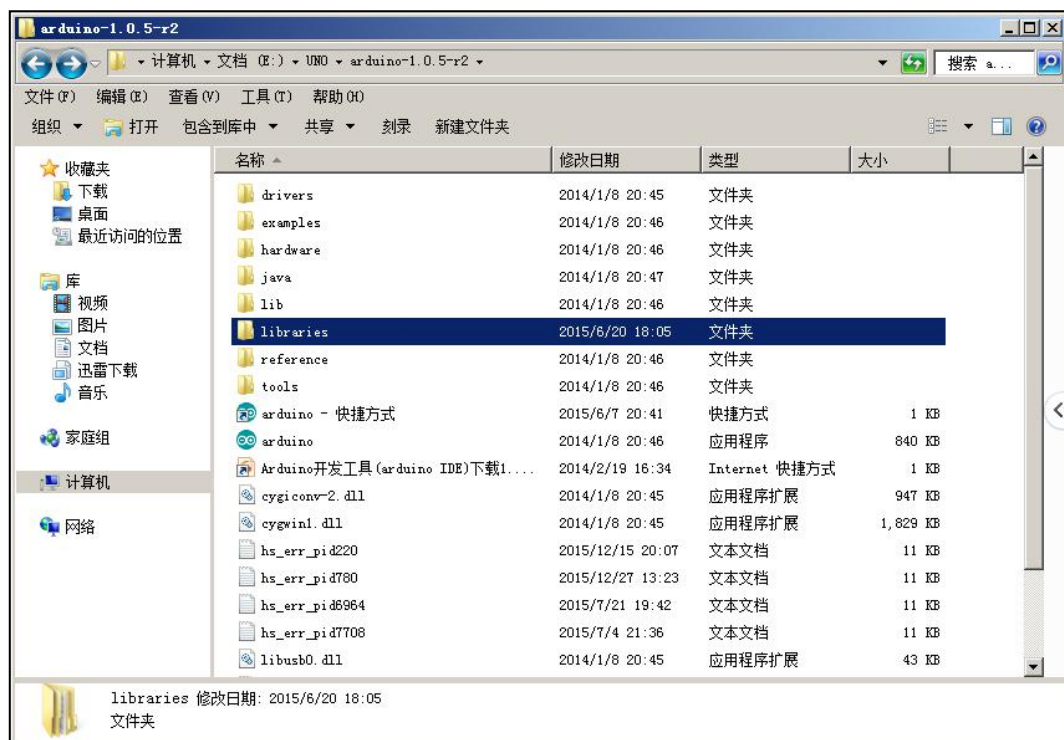


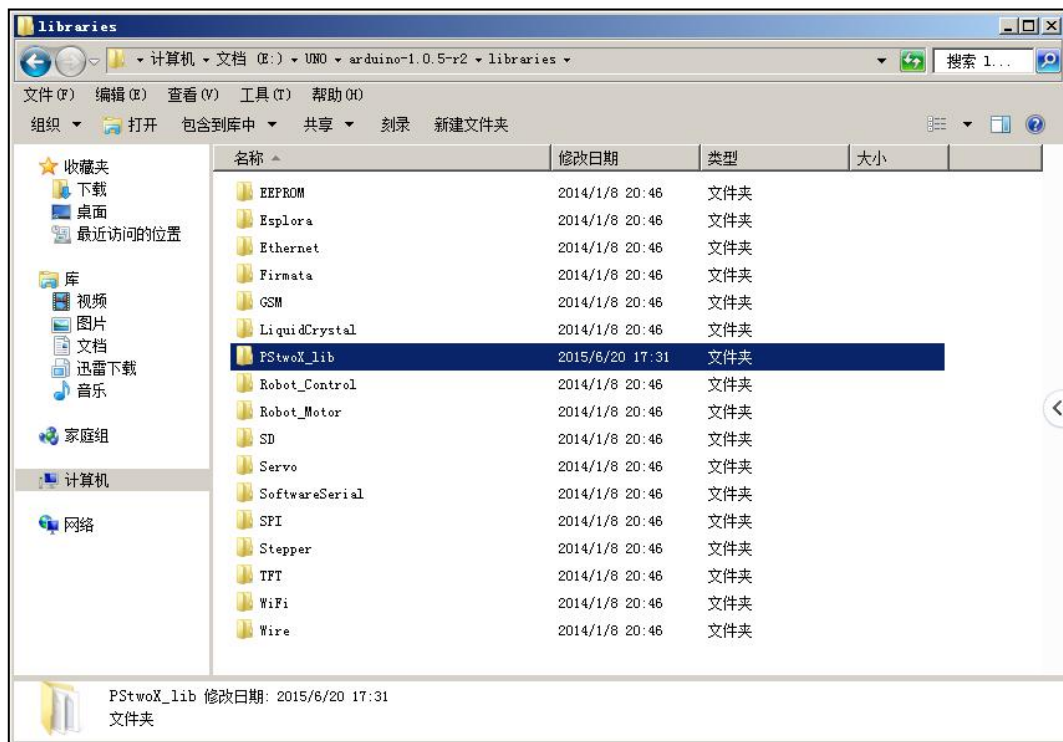
1、将《舵机转向小车用户手册》解压至电脑

找到文件夹 Libraries，将里面的 PS2 文件夹复制到 Arduino 库目录下

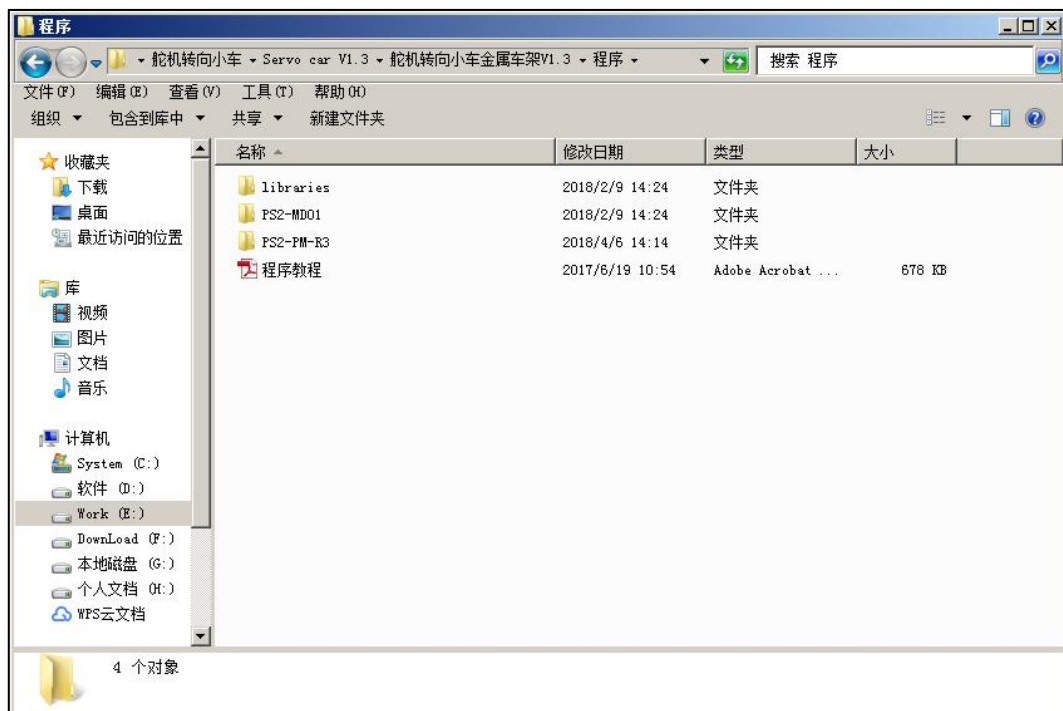


复制到下面文件夹中（所安装 IDE 盘）



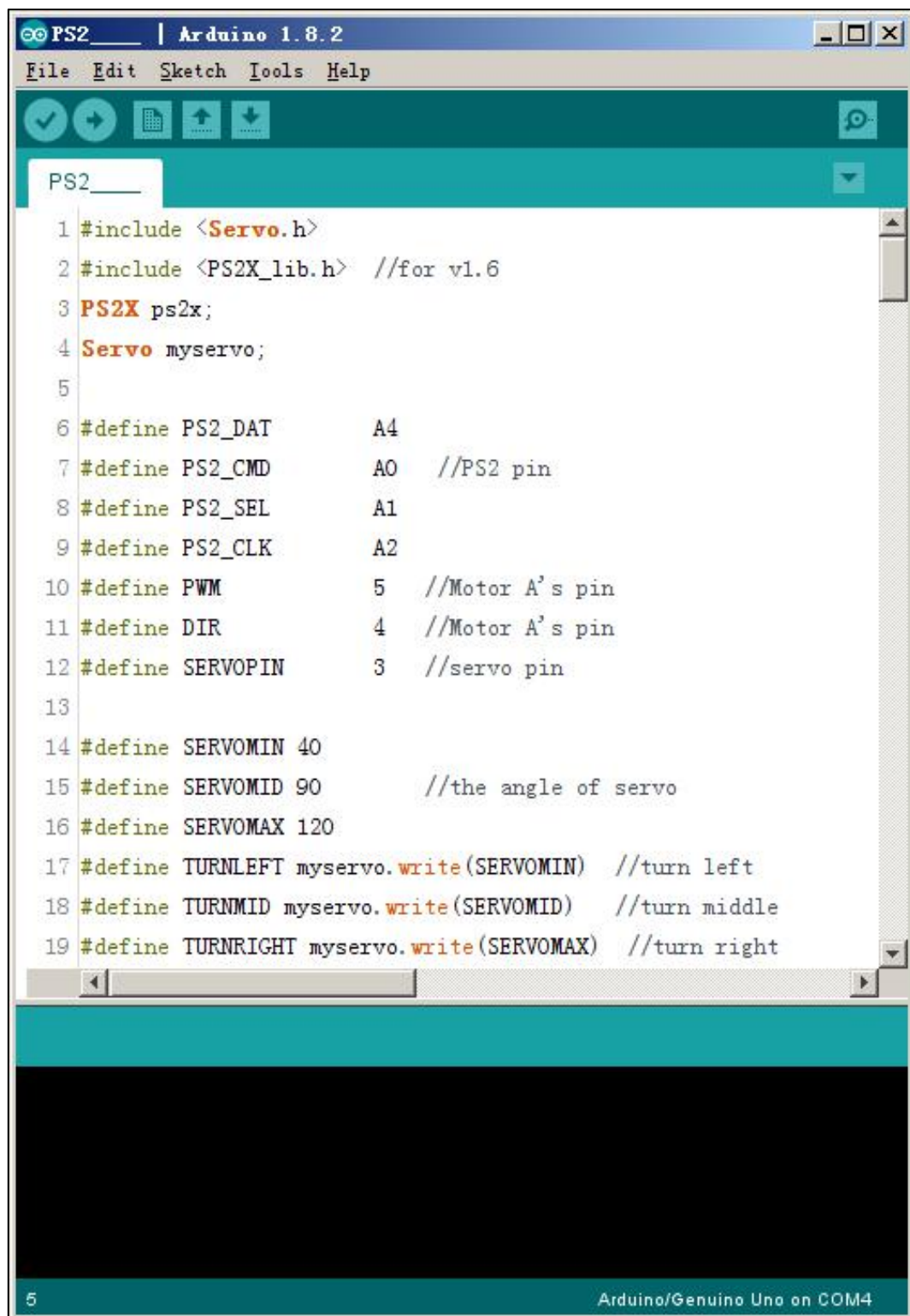


2、选择《程序》文件夹



- libraries:库文件
- PS2-MD01: 大功率型
- PS2-PM-R3: PM-R3 多功能扩展板型

打开程序如下图（PS2 程序）：



```
PS2____ | Arduino 1.8.2
File Edit Sketch Tools Help

PS2____
1 #include <Servo.h>
2 #include <PS2X_lib.h> //for v1.6
3 PS2X ps2x;
4 Servo myservo;
5
6 #define PS2_DAT      A4
7 #define PS2_CMD      A0 //PS2 pin
8 #define PS2_SEL      A1
9 #define PS2_CLK      A2
10 #define PWM          5 //Motor A's pin
11 #define DIR           4 //Motor A's pin
12 #define SERVOPIN     3 //servo pin
13
14 #define SERVOMIN 40
15 #define SERVOMID 90 //the angle of servo
16 #define SERVOMAX 120
17 #define TURNLEFT myservo.write(SERVOMIN) //turn left
18 #define TURNMID myservo.write(SERVOMID) //turn middle
19 #define TURNRIGHT myservo.write(SERVOMAX) //turn right

5 Arduino/Genuino Uno on COM4
```

程序的参数部分需要调整，下面介绍下每个程序代表的小车状态

```

PS2_
5
6 #define PS2_DAT      A4
7 #define PS2_CMD      A0  //PS2 pin
8 #define PS2_SEL      A1
9 #define PS2_CLK      A2
10 #define PWM          5  //Motor A's pin
11 #define DIR           4  //Motor A's pin
12 #define SERVO_PIN    3  //servo pin
13
14 #define SERVOMIN 40
15 #define SERVOMID 90      //the angle of servo
16 #define SERVOMAX 120
17 #define TURNLEFT myservo.write(SERVOMIN) //turn left
18 #define TURNMID myservo.write(SERVOMID)  //turn middle
19 #define TURNRIGHT myservo.write(SERVOMAX) //turn right
20 #define MOTORFORWARD setMotor(255) //motor forward
21 #define MOTORBACK setMotor(-255)    //motor back
22 #define MOTORSTOP setMotor(0)       //motor stop
23

```

1. PS2 接线引脚；修改引脚因遵循软硬件同步，不可单独修改程序或接线
2. 电机、舵机引脚；电机 PWM 由 5 引脚控制，方向 DIR 由 4 引脚控制，舵机引脚 3
3. 舵机初始范围；40 左、90 中、120 右
4. 电机速度，255 最快，0 最慢（停止）这里是按键控制小车速度，不包含摇杆

```

131 if (EnableRockerL) {
132   if (0 <= ps2x.Analog(PSS_LY) && ps2x.Analog(PSS_LY) <= 115) {
133     digitalWrite(DIR, LOW);
134     analogWrite(PWM, map(ps2x.Analog(PSS_LY), 0, 115, 255, 0));
135     LStop = 0;
136   } else if (255 >= ps2x.Analog(PSS_LY) && ps2x.Analog(PSS_LY) >= 140) {
137     digitalWrite(DIR, HIGH);
138     analogWrite(PWM, map(ps2x.Analog(PSS_LY), 140, 255, 0, 255));
139     LStop = 0;

```

- 红色框中是可以修改的值，这里是摇杆控制小车的速度，摇杆幅度越大，速度越快。
(0, 115, 255, 0) 代表将摇杆 0 到 115 值转化为电机 PWM 0 到 255；(0, 115) 是摇杆范围值 (255, 0) 是电机速度范围值，下同。
- 摇杆范围 (0-127、128-255) 由于摇杆精度问题，可能设置到 127 会导致小车通电转动，故这里选 115, 140，也可自己根据自己的手柄修改参数。


```

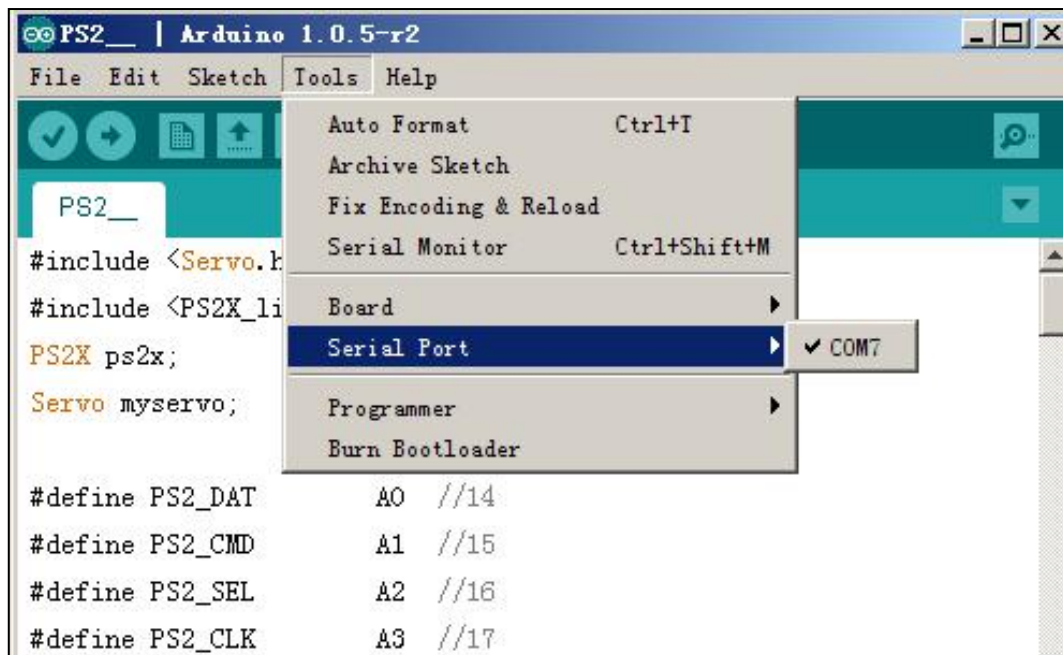
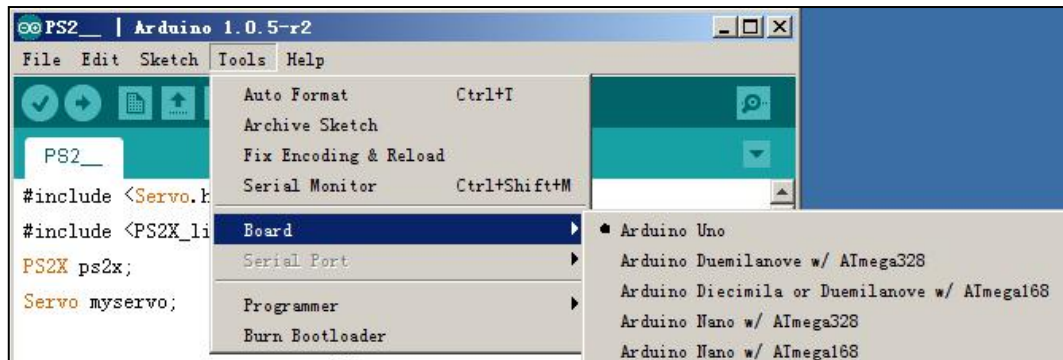
171 if (EnableRockerR) { //Left and Right
172   if (0 <= ps2x.Analog(PSS_RX) && ps2x.Analog(PSS_RX) <= 115) {
173     myservo.write(map(ps2x.Analog(PSS_RX), 0, 115, 40, 88));
174     RMid = 0;
175   } else if (255 >= ps2x.Analog(PSS_RX) && ps2x.Analog(PSS_RX) >= 140) {
176     myservo.write(map(ps2x.Analog(PSS_RX), 140, 255, 92, 120));
177     RMid = 0;

```

- 舵机角度范围（摇杆控制部分）红框中数值可以按实际情况修改（0, 115, 40, 88）：（0, 115）是摇杆范围值（40, 88）是舵机范围值，下同。
- 摇杆范围（0-127、128-255）由于摇杆精度问题，可能设置到 127 会导致小车通电舵机角度偏移转动，故选 115, 140，也可自己根据自己的手柄修改参数。

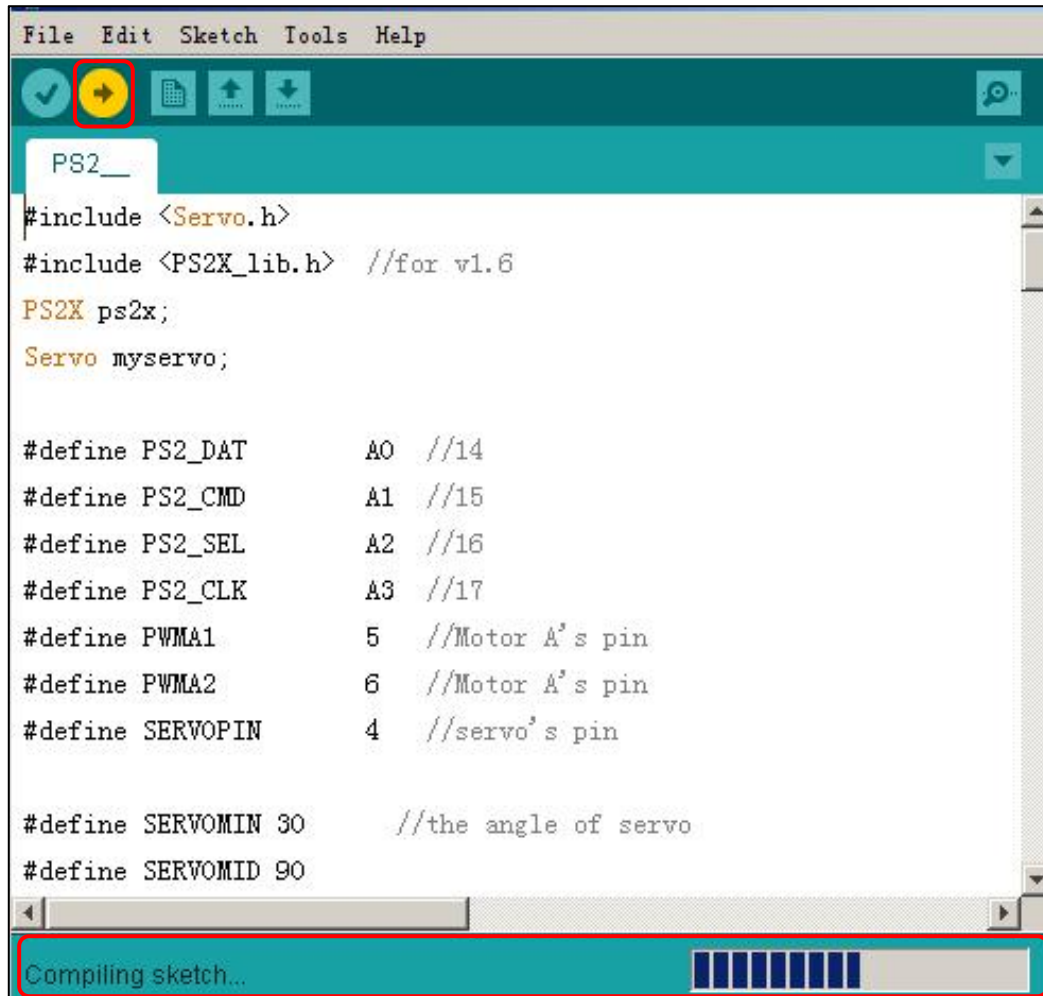
3: 下载程序到控制板中

下载之前先要选择控制板类型和端口（这里我们用的是蓝鸟主板，所以选择 UNO，如果端口没出现请检查驱动是否安装）

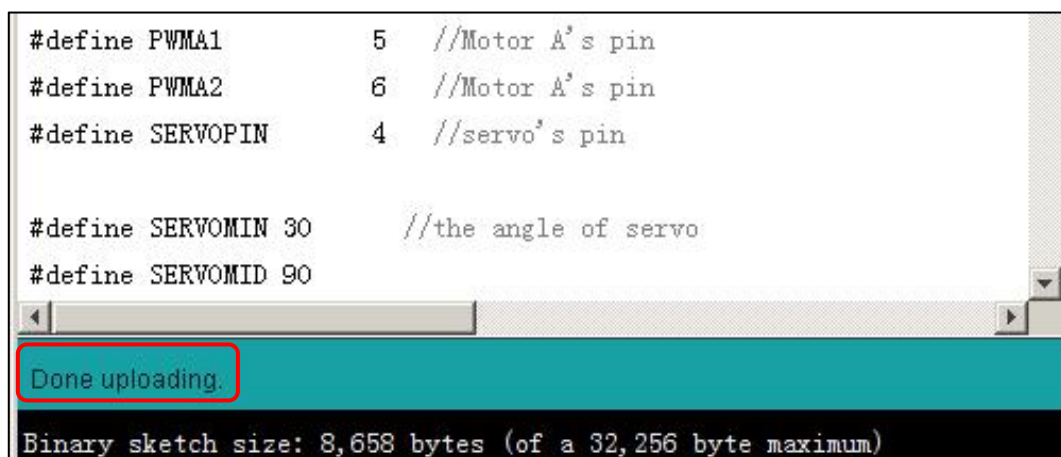


3-2: 选择好便开始下载程序到控制板中（图中黄色按钮）

正在下载（Compiling sketch）

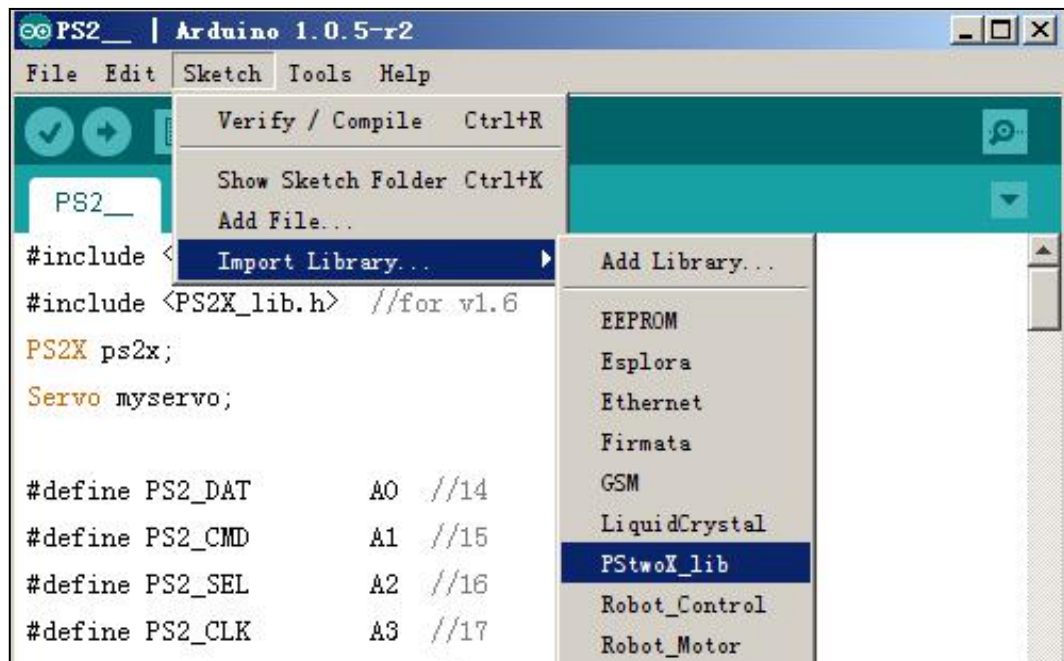


下载完成（Done uploading）



4: 导入库文件（导入的库文件必须在 Arduino 库文件夹里才可以导入）

下图示例是导入 PS2 库



导入舵机库

