



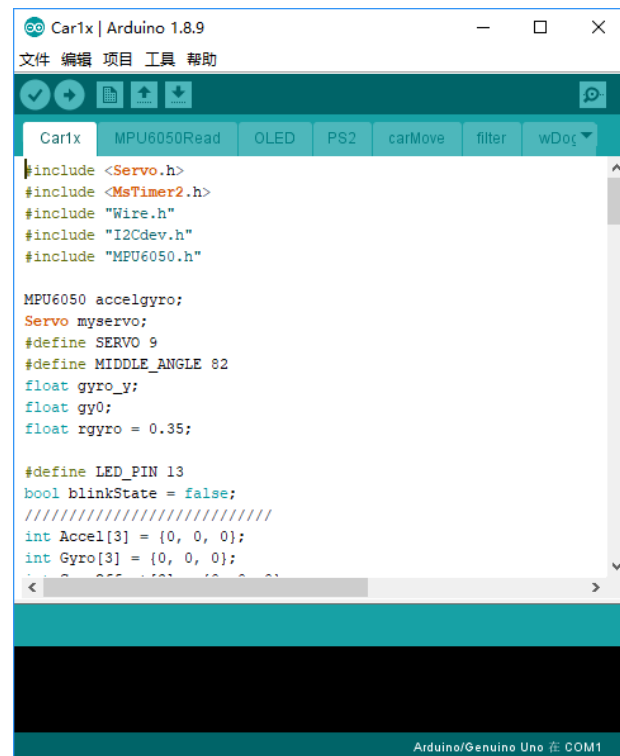
Arduino基础

什么是Arduino

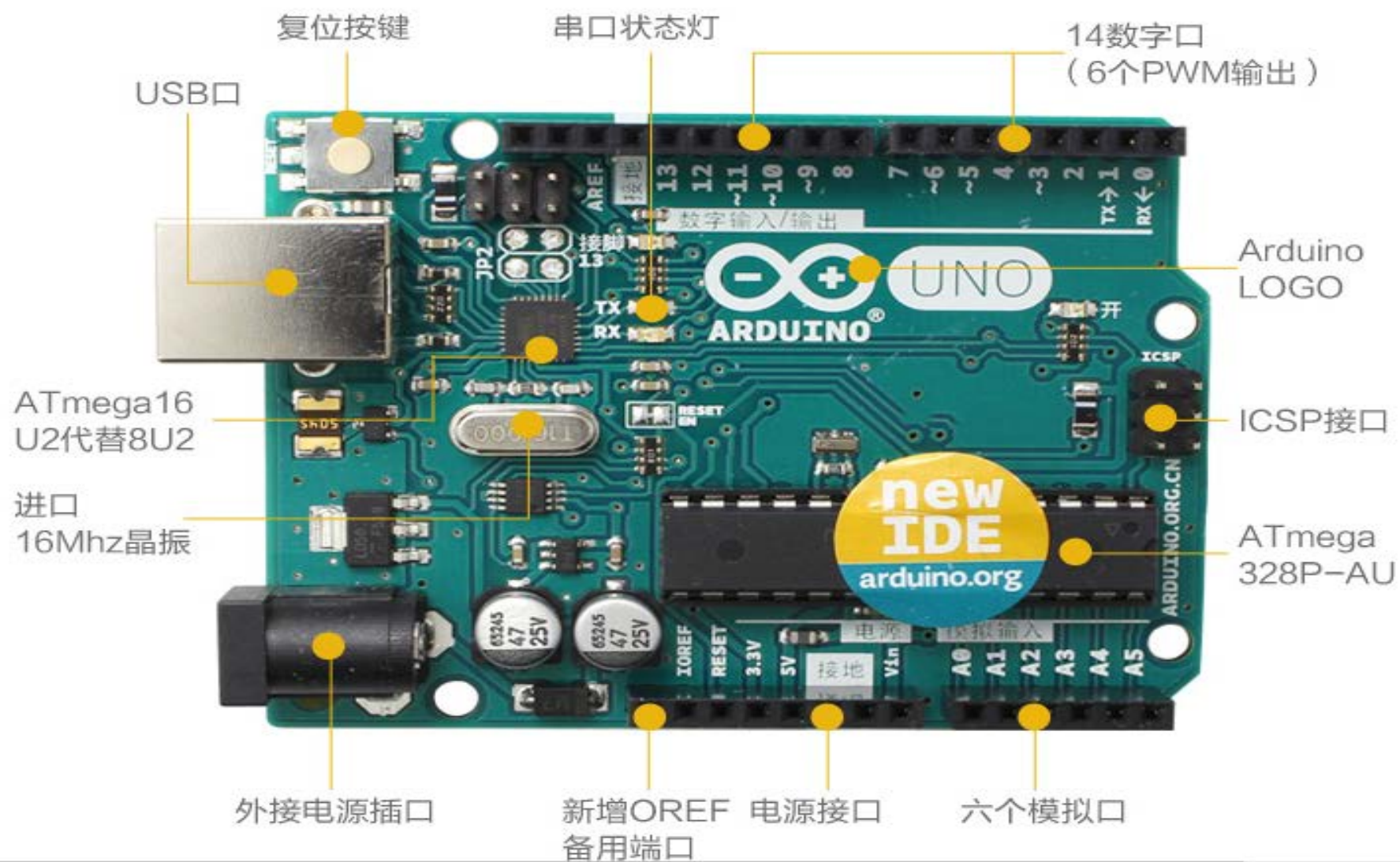
- Arduino由一个基于单片机并且开放源码的硬件平台和程序开发环境组成。可以用来开发交互产品，比如它可以读取大量的开关和传感器信号，并且可以控制各式各样的电机和其他物理设备。Arduino程序可单独运行，也可以在运行时和电脑中运行的程序进行通讯。

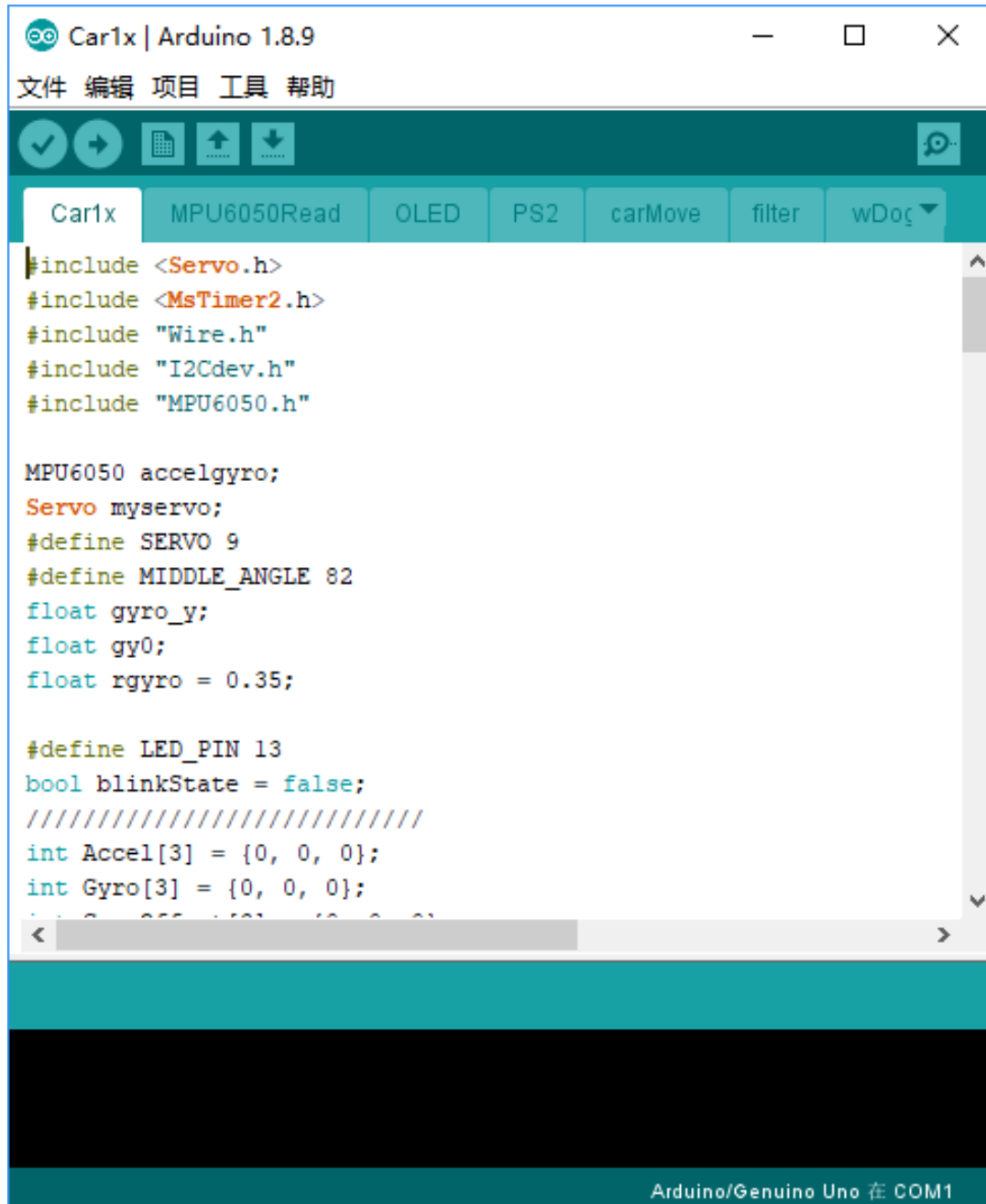


硬件平台——UNO开发板



集成开发环境（IDE）



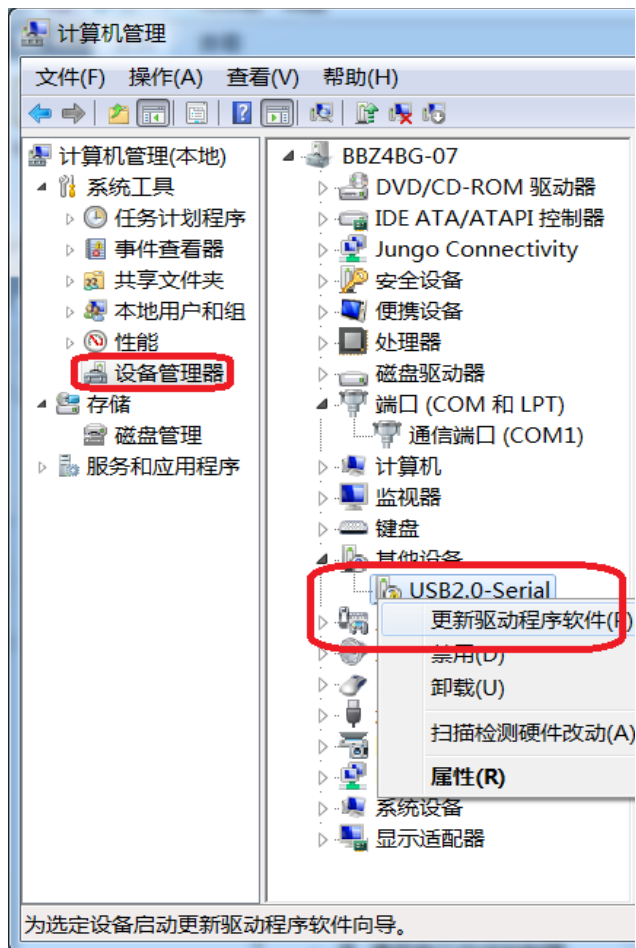


开发环境搭建

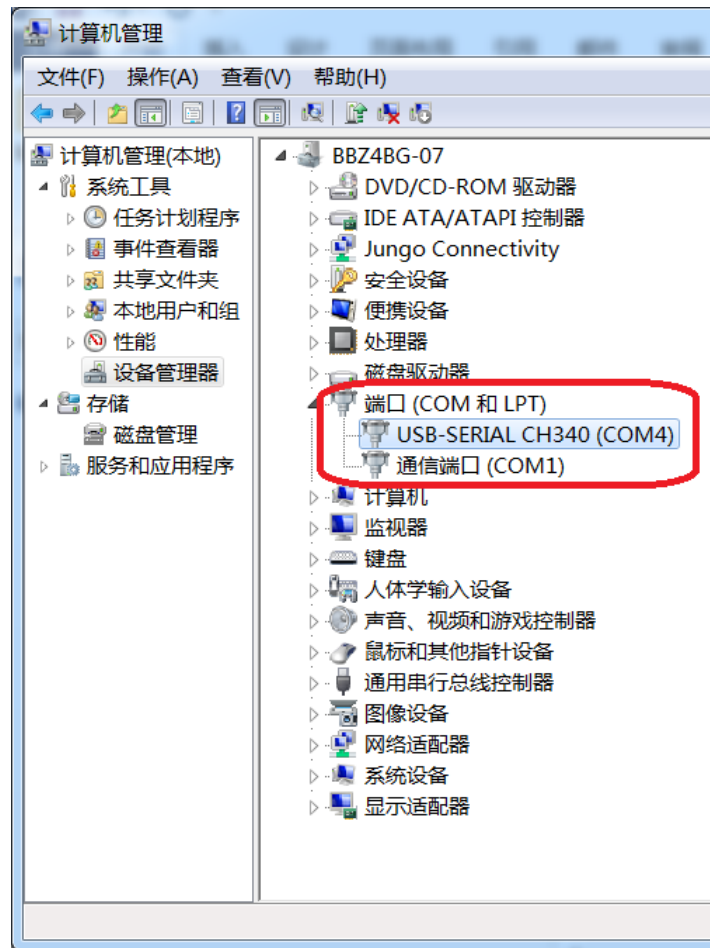
- 安装Arduino IDE
 - Arduino IDE可以从网上免费下载到
- 安装硬件平台驱动



Arduino IDE的图标



驱动安装前后UNO板在设备管理器中显示的名称



集成开发环境——界面



- ✓ 验证, 检查语法
- ➡ 编译, 编译、上传
- 📄 新建, 新建项目
- ⬆ 打开, 打开项目
- ⬇ 保存, 保存当前项目中所有文件
- 🔍 串口监视器, 打开串口监视器

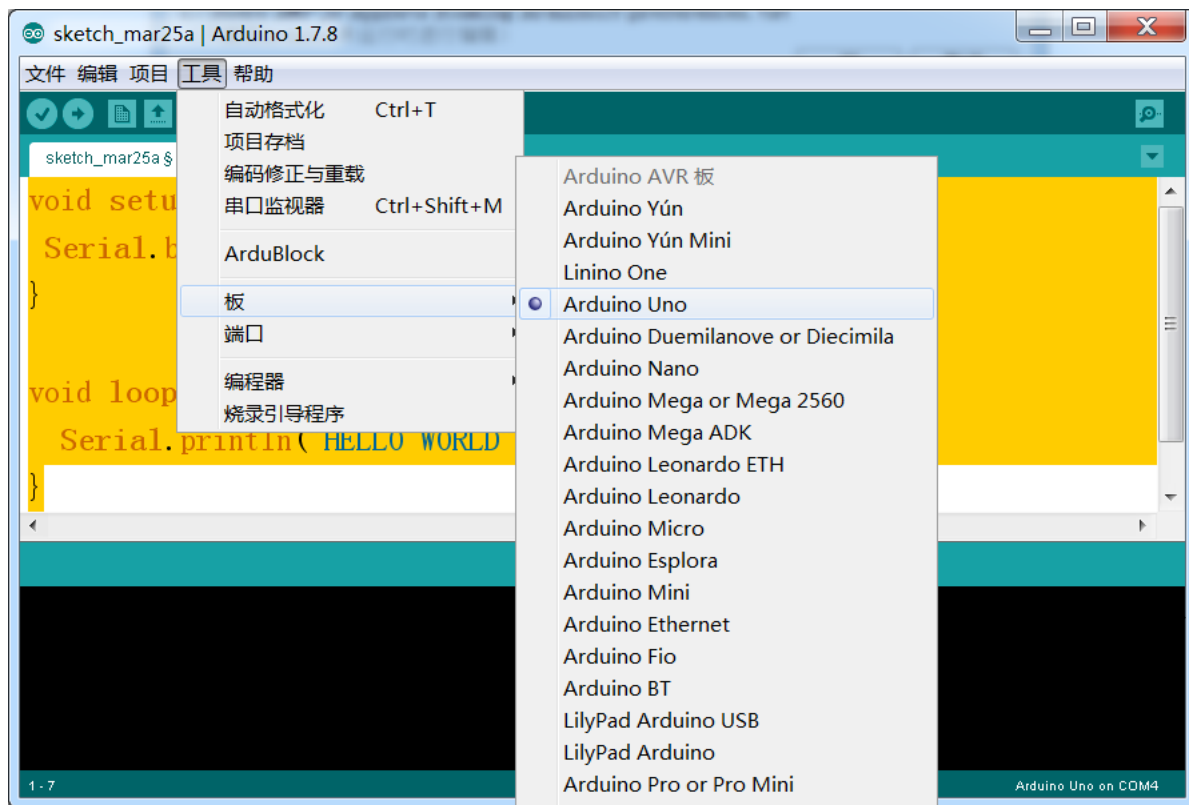
集成开发环境——导入库

- 使用第三方代码库可以加速项目的开发进度
- 添加.ZIP库...
- 选中库文件夹
- 点击打开
- 依次添加：
 - I2Cdev
 - MPU6050
 - MsTimer2-master
 - PS2X_lib

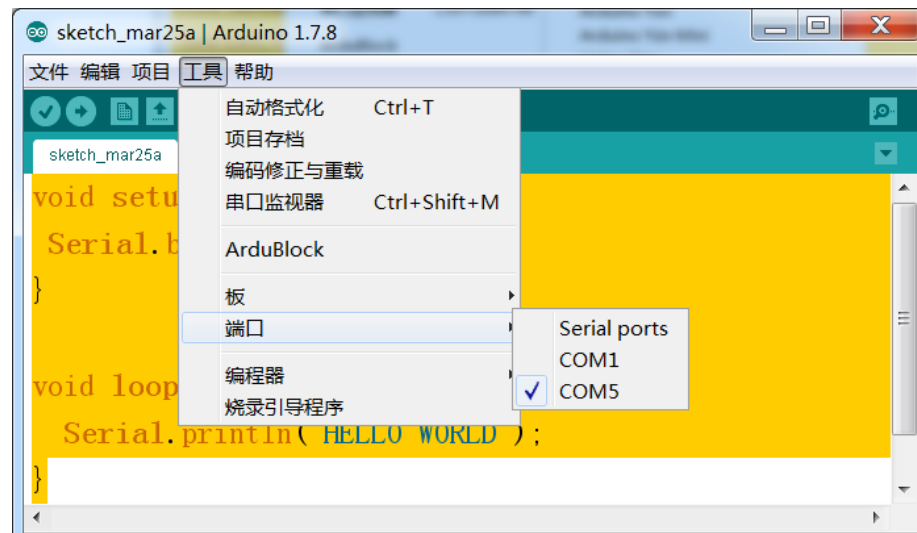


集成开发环境——上传

- 上传是将编译好的程序发送到UNO板中执行



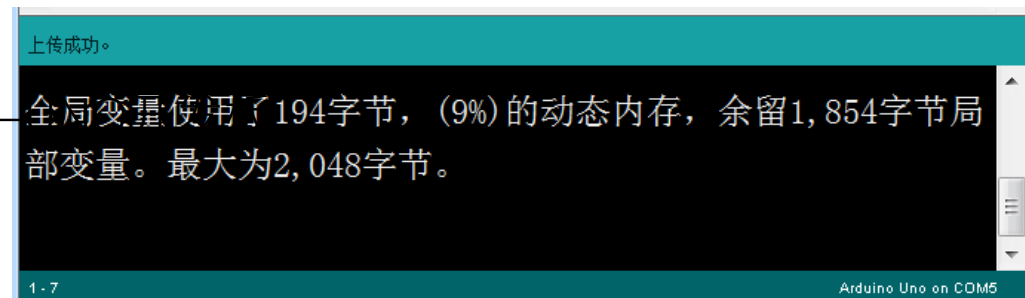
1.选择使用的开发板



2.选择通讯端口



3.编译、上传



4.显示上传成功



Arduino程序设计基础

什么是程序

- 一组**按照一定次序执行的**，计算机能够读懂的**指令**
- 存储程序原理
 - 将指令一条一条的存储在存储器中，CPU从存储器中读出指令并**依次运行**

```
while(1)           //下面大括号里的程序不停的循环
{
    LED_ON();       //LED灯打开
    delay(1000);    //等待1秒
    LED_OFF();      //LED灯熄灭
    delay(1000);    //等待1秒
}
```



举个栗子 读一读，看看能不能懂

初识Arduino： 下载： 08_1_HelloWorld

```
int Val=0;
void setup()
{
  // put your setup code here, to run once:
  Serial.begin(9600);//串口初始， 设置波特率
  Serial.println("This is my work,my name is ???");//从串口输出调试
  信息
}
```

初识Arduino： 下载： 08_1_HelloWorld

```
void loop()
{
  // put your main code here, to run repeatedly:
  Val=Serial.read();//读取PC 机发送给Arduino 的指令或字符，并将该指令或字符赋给val
  if(Val=='D')//判断接收到的指令或字符是否是“D”。
  {//如果接收到的是“D”字符
    Serial.println("Today is Wednesday");
  }
  if(Val=='W')//判断接收到的指令或字符是否是“W”。
  {//如果接收到的是“W”字符
    Serial.println("It is suny");
  }
}
```

初识Arduino

下载：08_1>HelloWorld

改改看看！

实践5分钟~~~~~

基本框架

setup函数

函数中的语句会在单片机上电后第一时间运行，但只会运行一次，就像是windows的启动动画

loop函数

函数中的语句会在Setup函数执行完后开始执行；执行完所有的语句后会回到loop的第一句话再次运行，循环往复，直到断电。

函数

就是将一系列的语句包装起来，形成一个特定的功能包

注释

“//”后面的语句对程序没有影响，是用来给读程序的人看的，起解释或者提示作用。被“/*”和“*/”包夹起来的内容也是注释

语句

它调用了一个系统库函数，打开串口，并把串口速度设置为9600bps
语句的结尾要加“;”

```
void setup() {  
    // put your setup code here, to run once:  
    Serial.begin(9600); // 串口初始，设置波特率  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    Serial.print("Hello "); // 从串口输  
    Serial.println("World!"); // 从串口输出后换行  
}
```

语句

调用系统库函数，从串口向电脑发出World!字符串和回车

语句

调用系统库函数，从串口向电脑发出Hello 字符串



跑一跑这段程序，看看有什么结果

数字IO控制

- IO是输入/输出接口的简称
- UNO板上0-13是数字IO口
- A0-A5则是允许模拟输入的IO口
- 数字IO口的使用
 - 首先要设置它是输入还是输出
 - 如果是输入，则需要使用系统库函数读取IO口的高低电平（0v还是5v）
 - 如果是输出，则需要使用系统库函数设定IO口的高低电平（0v还是5v）

系统库函数
设置引脚模式

第一个输入参数
要设定的引脚编号
0-13, A0-A5

第二个输入参数
输入、输出
INPUT/OUTPUT

```
pinMode(13, OUTPUT);
```

用于存放读取
到数据的变量

系统库函数
读取数字输入
引脚电平

输入参数
要读取的引脚编号
0-13, A0-A5

```
val = digitalRead(7);
```

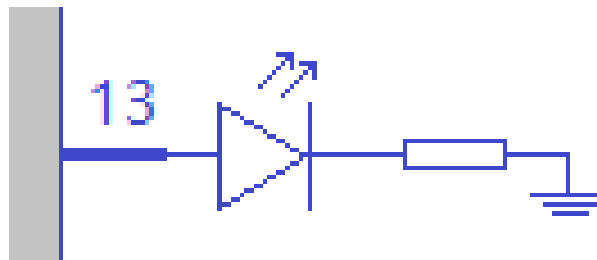
系统库函数
设置数字输出
引脚电平

第一个输入参数
要设定的引脚编号
0-13, A0-A5

第二个输入参数
高电平、低电平
HIGH/LOW

```
digitalWrite(13, LOW);
```

LED控制

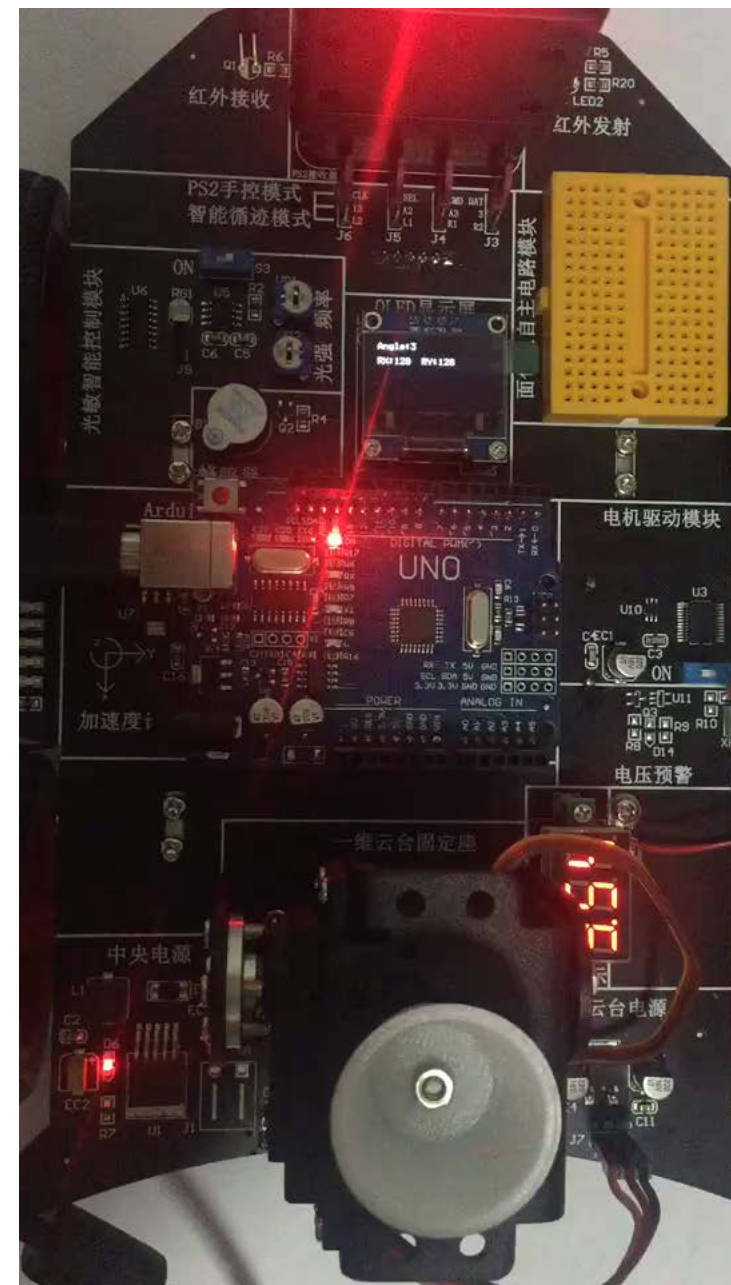


```
void setup()
{
  pinMode(13, OUTPUT);      //将 13 号数字口设置为输出
}
void loop()
{
  digitalWrite(13, HIGH);   // 点亮 LED 灯 (HIGH 表示数字引脚输出高电平)
  delay(1000);              // 等待 1000 毫秒, 即等待 1 秒
  digitalWrite(13, LOW);    // 熄灭 LED 灯 (LOW 表示数字引脚输出低电平)
  delay(1000);              // 等待 1000 毫秒, 即等待 1 秒
}
```

读一读这段程序, 想想运行会有什么结果

下载: 08_2_Blink
改改看看!

实践5分钟~~~~~



蜂鸣器控制

- 硬件电路

- 蜂鸣器所在IO 12
- 高电平发声

宏定义

以#开头;
以后的程序中, 凡是
见到Beep就与12等效;
宏定义后面不加";"

for循环

for后面大括号中的语句将被循环运行;
i=0: 循环变量i初始值为0,
i++: 每次循环中i+1,
i<5: 在i<5时进行循环
(当i>=5时, 退出循环);
for后面不加";"

```
#define Beep 12 //定义蜂鸣器所在IO
void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600); //串口初始, 设置波特率
    pinMode(Beep, OUTPUT); //设置蜂鸣器所在IO输出
    digitalWrite(Beep, LOW); //初始化蜂鸣器不发声
}

void loop() {
    // put your main code here, to run repeatedly:
    Serial.print("Beep:B"); //从串口输出调试信息
    digitalWrite(Beep, HIGH); //打开蜂鸣器
    for(int i=0; i<5; i++) //循环5次
    {
        Serial.print("e"); //从串口输出调试信息
        delay(400); //延时400ms
    }
    Serial.println("p!"); //从串口输出后换行
    digitalWrite(Beep, LOW); //关闭蜂鸣器
    delay(2000); //延时2000ms
}
```

牛刀小试



“?”是你需要自己填写的内容

- 通过电脑上的串口监视器输入控制字符;
- 当输入字符“H”时, 让蜂鸣器鸣笛, LED 点亮;
- 当输入字符“L”时, 让蜂鸣器熄灭, LED 熄灭。

返回从串口收到的数据

If判断

if后面小括号中判断如果成立,
下方大括号中的语句才会执行;
两个等号用于判断是否相等;
单引号中加字母代表一个字符;
if后面不加“;”

```
#define Beep ? //定义蜂鸣器所在 IO
void setup()
{
    // put your setup code here, to run once:
    Serial.begin(?); //串口初始, 设置波特率
    pinMode(Beep, ?); //设置蜂鸣器所在 IO 输出
    digitalWrite(Beep, ?); //初始化蜂鸣器不发声
    Serial.print("This is my work, my name is ???"); //从串口输出调试信息
}
void loop()
{
    // put your main code here, to run repeatedly:
    int Val;
    Val = Serial.read(); //读取 PC 机发送的指令或字符, 并将该指令或字符赋给 val
    if (Val == 'H') //判断接收到的指令或字符是否是“H”。
    { //如果接收到的是“H”字符
        Serial.println("OK, Sir, Beep will be on");
        digitalWrite(Beep, ?); //打开蜂鸣器
    }
    if (Val == 'L') //判断接收到的指令或字符是否是“L”。
    { //如果接收到的是“L”字符
        Serial.println("OK, Sir, Beep will be off");
        digitalWrite(Beep, ?); //打开蜂鸣器
    }
}
```


下载： [08_11_CtlBeep](#)

补充缺失部分，看看效果~

蜂鸣器

