**上海交通大学黑客马拉松**

**Arduino使用指南**

**Hackathon OMNI-Lab**

目录

[一、Arduino简介 3](#_Toc398398623)

[1．1、什么是Arduino 3](#_Toc398398624)

[1. 2、Arduino特点 3](#_Toc398398625)

[1. 3、Arduino性能 3](#_Toc398398626)

[1. 4、Arduino技术指标 3](#_Toc398398627)

[1.4.1 概要 3](#_Toc398398628)

[1.4.2 电源 4](#_Toc398398629)

[1.4.3 输入输出 4](#_Toc398398630)

[1.4.4 通信接口 5](#_Toc398398631)

[二、搭建Arduino环境 5](#_Toc398398632)

[1、安装Arduino 5](#_Toc398398633)

[2、安装USB驱动 5](#_Toc398398634)

[三、Arduino 语言介绍 11](#_Toc398398635)

[3.1、Arduino程序基本构架 11](#_Toc398398636)

[3.1.1 Arduino 程序结构 11](#_Toc398398637)

[3.1.2 Arduino 函数功能 11](#_Toc398398638)

[3.2、Arduino常用编辑语言 13](#_Toc398398639)

[3.2.1 关键词 13](#_Toc398398640)

[3.2.2语法符号 14](#_Toc398398641)

[3.2.3运算符 14](#_Toc398398642)

[3.2.4 数据类型 14](#_Toc398398643)

[3.2.5数据类型转换 15](#_Toc398398644)

[3.2.6常量 15](#_Toc398398645)

[四、Arduino实例 15](#_Toc398398646)

[4.1、准备工作 15](#_Toc398398647)

[4.1.1、Arduino环境 15](#_Toc398398648)

[4.2、串口监视器使用---“Hello World!” 18](#_Toc398398649)

[4.3、外设传感器使用---点亮普通LED 23](#_Toc398398650)

[4.4、外设传感器使用---点亮双色LED 24](#_Toc398398651)

[4.5、函数库的调用---DHT22温湿度传感器 24](#_Toc398398652)

# 一、Arduino简介

## 1．1、什么是Arduino

Arduino，是一个基于开放原始码的软硬件平台，构建于开放原始码simple I/O介面版，并且具有使用类似Java，C语言的Processing/Wiring开发环境。 Arduino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灯、电机和其他物理设备。Arduino项目可以是单独的，也可以在运行时和你电脑中运行的程序（例如：Flash，Processing，MaxMSP）进行通讯。Arduino板你可以选择自己去手动组装或是购买已经组装好的；Arduino开源的IDE可以免费下载得到。 Arduino的编程语言就像似在对一个类似于物理的计算平台进行相应的连线，它基于处理多媒体的编程环境。

## 1. 2、Arduino特点

1、开放[源代码](http://baike.baidu.com/view/60376.htm)的电路图设计，[程序](http://baike.baidu.com/view/17674.htm)开发接口[免费](http://baike.baidu.com/view/2763.htm)下载，也可依个人需求自己修改。

2、使用低价格的微处理[控制器](http://baike.baidu.com/view/122229.htm)(AVR系列控制器)，可以采用[USB接口](http://baike.baidu.com/view/26083.htm)供电，不需外接电源，也可以使用外部9VDC输入。

3、Arduino支持ISP在线烧，可以将新的“bootloader”[固件](http://baike.baidu.com/view/33738.htm)烧入AVR芯片。有了bootloader之后，可以通过[串口](http://baike.baidu.com/view/69108.htm)或者USB to Rs232线更新[固件](http://baike.baidu.com/view/33738.htm)。

4、可依据官方提供的Eagle格式PCB和SCH电路图简化Arduino模组，完成独立运作的微处理控制；可简单地与传感器，各式各样的电子元件连接(例如：红外线,超音波,热敏电阻,光敏电阻,伺服马达,…等)

5、支持多种互动程序，如：Flash、Max/Msp、[VVVV](http://baike.baidu.com/view/777692.htm)、PD、C、Processing等。

6、应用方面，利用Arduino，突破以往只能使用[鼠标](http://baike.baidu.com/view/2199.htm)、[键盘](http://baike.baidu.com/view/7402.htm)、CCD等输入的装置的互动内容，可以更简单地达成单人或多人游戏互动。

## 1. 3、Arduino性能

Arduino Mega2560也是采用USB接口的核心电路板，它最大的特点就是具有多达54路数字输入输出，特别适合需要大量IO接口的设计。Mega2560的处理器核心是[ATmega2560](http://kb.open.eefocus.com/index.php/ATmega2560)， 同时具有54路数字输入/输出口（其中16路可作为PWM输出），16路模拟输入，4路UART接口，一个16MHz晶体振荡器，一个USB口，一个电源 插座，一个ICSP header和一个复位按钮。Arduino Mega2560也能兼容为Arduino UNO设计的扩展板。

## 4、Arduino技术指标

## 1.4.1 概要

* 处理器 ATmega2560
* 工作电压 5V
* 输入电压（推荐） 7-12V
* 输入电压（范围） 6-20V
* 数字IO脚 54 (其中16路作为PWM输出）
* 模拟输入脚 16
* IO脚直流电流 40 mA
* 3.3V脚直流电流 50 mA
* Flash Memory 256 KB （ATmega328，其中8 KB 用于 bootloader）
* SRAM 8 KB
* EEPROM 4 KB
* 工作时钟 16 MHz

## 1.4.2 电源

Arduino Mega2560可以通过3种方式供电，而且能自动选择供电方式

* 外部直流电源通过电源插座供电。
* 电池连接电源连接器的GND和VIN引脚。
* USB接口直接供电。

电源引脚说明

* VIN --- 当外部直流电源接入电源插座时，可以通过VIN向外部供电；也可以通过此引脚向Mega2560直接供电；VIN有电时将忽略从USB或者其他引脚接入的电源。
* 5V --- 通过稳压器或USB的5V电压，为UNO上的5V芯片供电。
* 3.3V --- 通过稳压器产生的3.3V电压，最大驱动电流50mA。
* GND --- 地脚。

## 1.4.3 输入输出

1. 14路数字输入输出口：工作电压为5V，每一路能输出和接入最大电流为40mA。每一路配置了20-50K欧姆内部上拉电阻（默认不连接)。除此之外，有些引脚有特定的功能

* 4路串口信号：串口0---0(RX)and 1(TX);串口1---19(RX)and 18(TX);串口2---17(RX)and 16(TX);串口3---15(RX)and 14(TX)。其中串口0与内部 ATmega8U2 USB-to-TTL 芯片相连，提供TTL电压水平的串口接收信号。
* 6路外部中断：2(中断0)，3(中断 1)，18(中断 5)，19(中断 4)，20(中断 3)，and 21(中断 2)。触发中断引脚，可设成上升沿、下降沿或同时触发。
* 14路脉冲宽度调制PWM（0--13）：提供14路8位PWM输出。
* SPI（53(SS)，51(MOSI)，50(MISO)，52(SCK)）：SPI通信接口。
* LED（13号）：Arduino专门用于测试LED的保留接口，输出为高时点亮LED，反之输出为低时LED熄灭。

1. 16路模拟输入：每一路具有10位的分辨率（即输入有1024个不同值），默认输入信号范围为0到5V，可以通过AREF调整输入上限。除此之外，有些引脚有特定功能

* [TWI](http://kb.open.eefocus.com/index.php?title=TWI&action=edit&redlink=1)接口（20（SDA）和21（SCL））：支持通信接口（兼容I2C总线）。

1. AREF：模拟输入信号的参考电压。
2. Reset：信号为低时复位单片机芯片。

## 1.4.4 通信接口

1. 串口：ATmega2560内置的4路UART可以与外部实现串口通信；ATmega16U2可以访问串口0实现USB上的虚拟串口。
2. 支持TWI（兼容I2C）接口。
3. 支持SPI 接口。

# 二、搭建Arduino环境

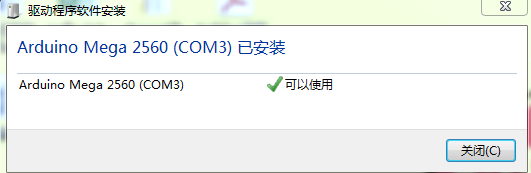
## 1、安装Arduino

采用Arduino IDE开发工具Arduino-1.0。5-r2 免安装开发工具。打开文件夹arduino-1.0.5-r2下的arduino.exe即可使用Arduino IDE。

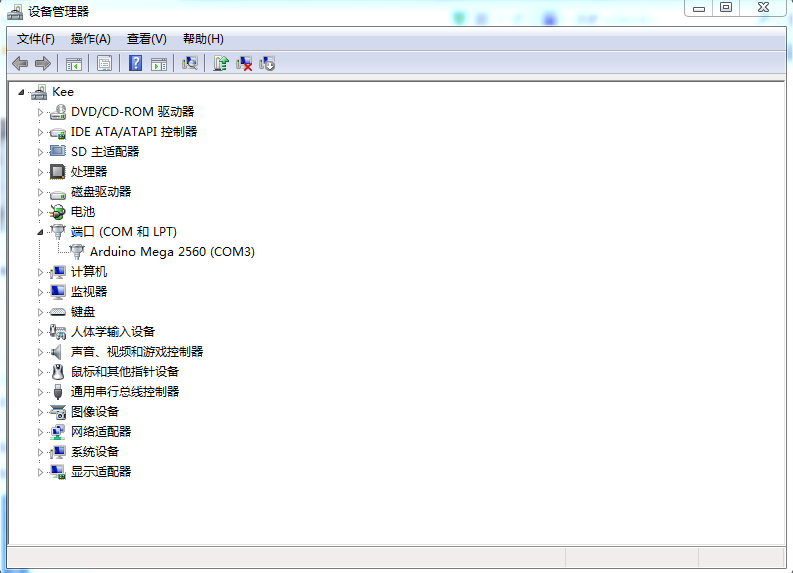
## 2、安装USB驱动

安装USB的驱动如下：

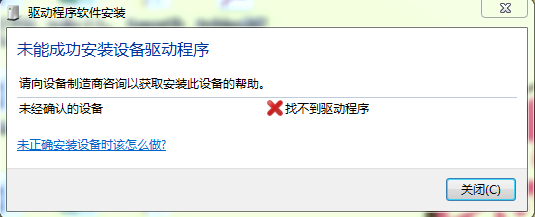
1. 连接Arduino开发板和电脑
2. 系统会自动检查设备，如果显示驱动安装成功，即可使用



此时可右击 计算机>属性> 左击“设备管理器”> 端口 查看端口情况确认驱动安装成功

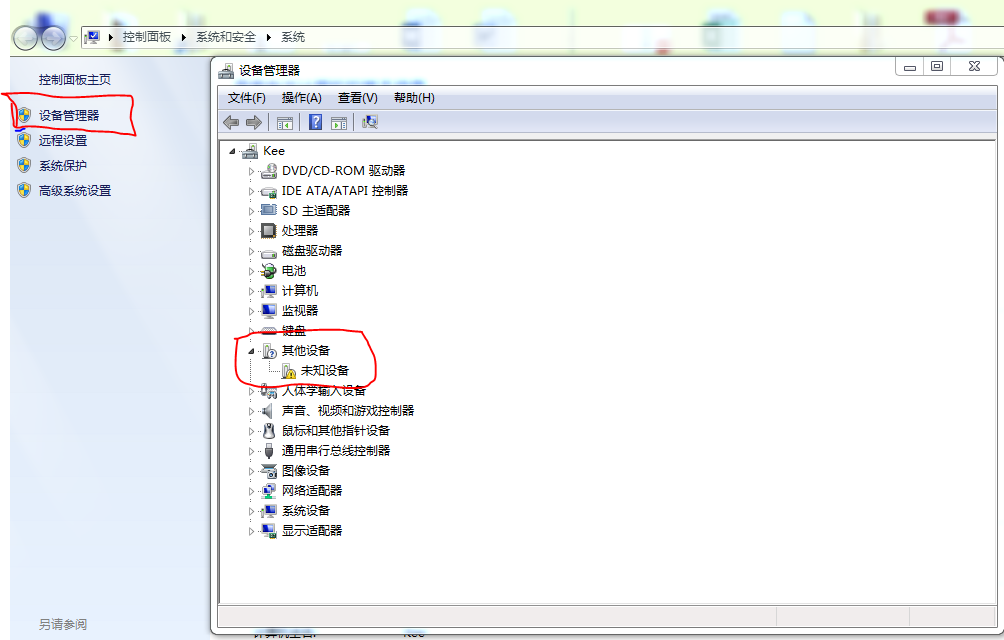


1. 如果出现设备安装不成功情况，

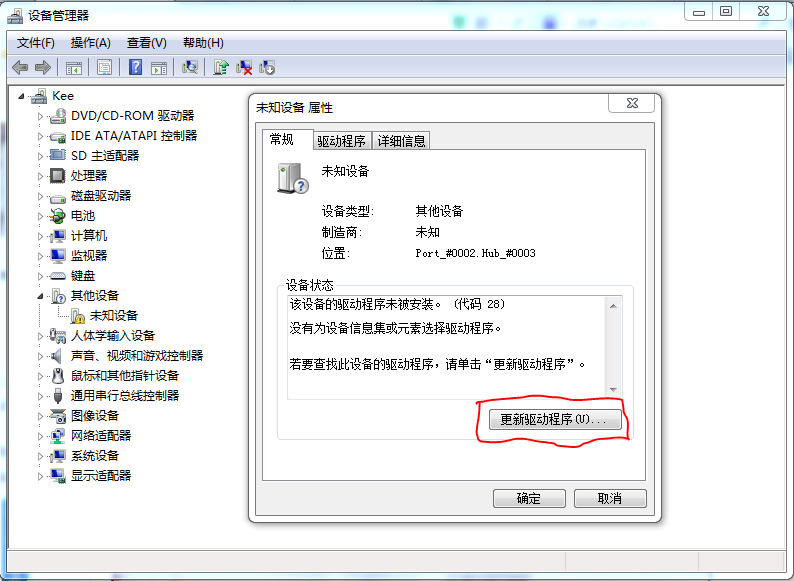


此时则需要手动安装驱动，步骤如下：

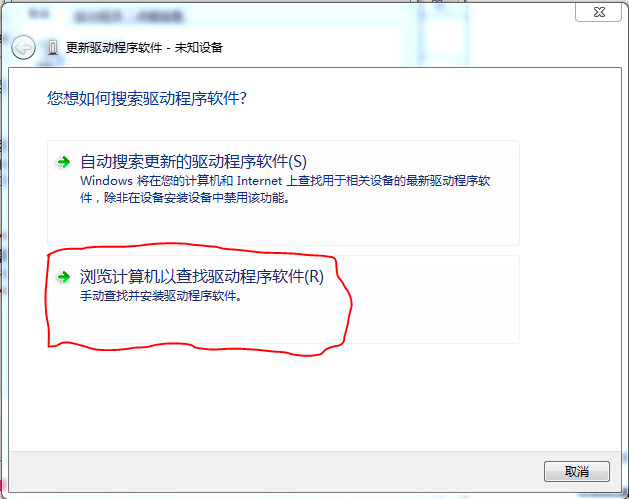
右键计算机选择“属性” >> 左击“设备管理器”> >在其他设备下 双击 “未知设备”，



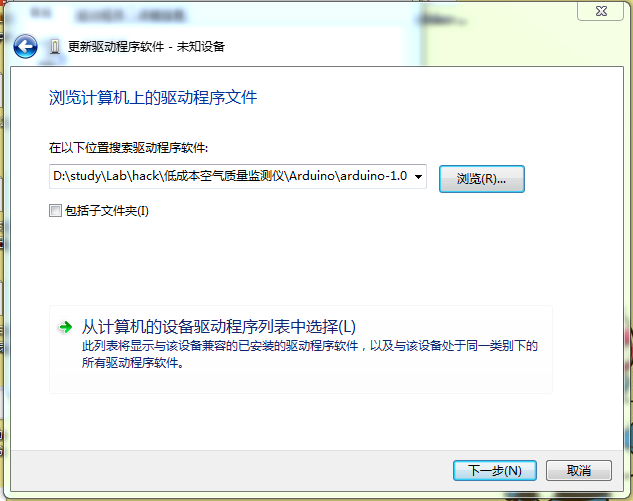
此时会出现“未知设备 属性”对话框，点击“更新驱动程序”

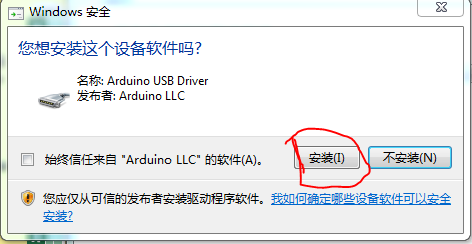


在出现的对话框中选择“连蓝计算机以查找驱动程序软件”



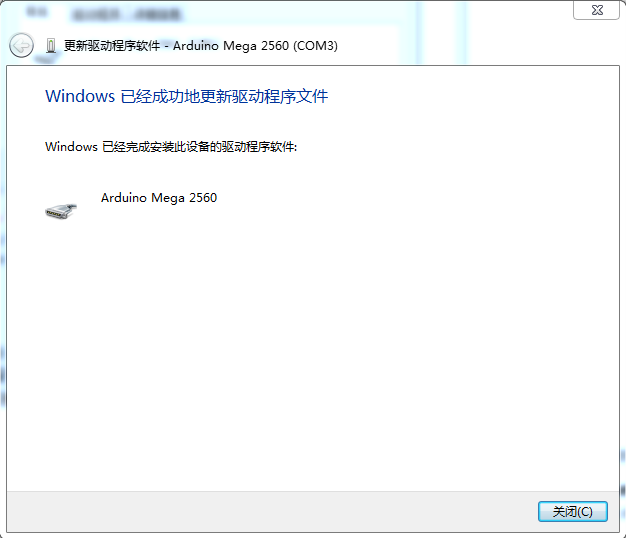
在出现的对话框中点击“浏览”找到arduino-1.0.5-r2安装文件夹下的drivers 添加到路径里，然后点击“下一步”

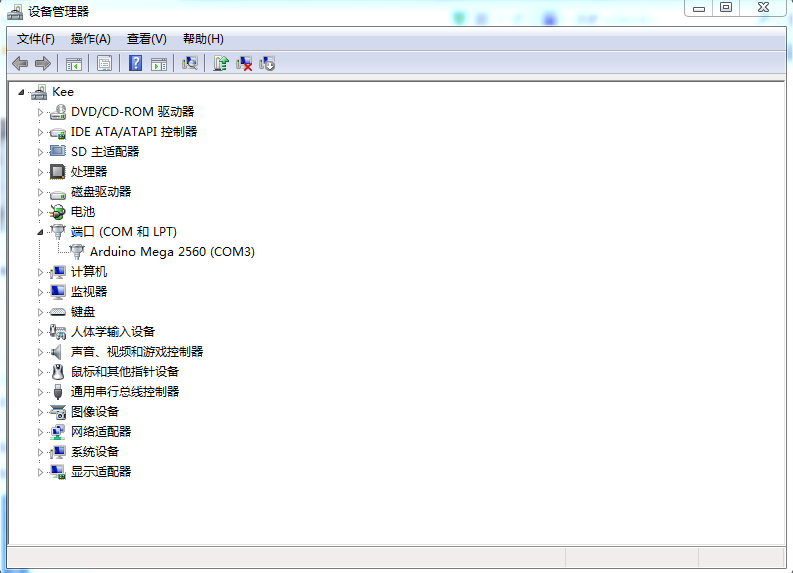




然后点击“安装”

安装完成后出现如下提示，显示驱动安装成功





# 三、Arduino 语言介绍

## 3.1、Arduino程序基本构架

### 3.1.1 Arduino 程序结构

1、声明变量及接口名称（int val;int ledPin=13;），调用库函数

2、setup()——函数在程序开始时使用，可以初始化变量、接口模式、启用库等（例如：pinMode(ledPin,OUTUPT);）

3、loop（）——在 setup()函数之后，即初始化之后，loop() 让你 的程序循环地被执行。使用它来运转 Arduino。

### 3.1.2 Arduino 函数功能

**数字 I/O：**

* [**pinMode**](http://arduino.cc/en/Reference/PinMode)(pin, mode)：数字IO口输入输出模式定义函数，pin表示为 0～53， mode表示为INPUT或OUTPUT。
* **Void** [**digitalWrite**](http://arduino.cc/en/Reference/DigitalWrite)(pin, value)：数字IO口输出电平定义函数，pin表示为 0～53，value表示为HIGH或LOW。 比如定义HIGH可以驱动LED。
* **int** [**digitalRead**](http://arduino.cc/en/Reference/DigitalRead)(pin)：数字IO口读输入电平函数，pin表示为0～53，value表示为HIGH或LOW。比如可以读数字传感器。

**模拟 I/O：**

* **int** [**analogRead**](http://arduino.cc/en/Reference/AnalogRead)(pin)：模拟IO口读函数，pin表示为 0～15比如可以读模拟传感器（10 位AD，0～5V表示为 0～1023）
* **Void** [**analogWrite**](http://arduino.cc/en/Reference/AnalogWrite)(pin, value) – *PWM：*数字IO口PWM输出函数，Arduino数字IO口标注了PWM的IO口可使用该函数，pin表示2～13，value表示为 0～255。比如可用于电机PWM调速或音乐播放、LED变色控制等。

**时间函数：**

* **unsigned long** [**millis**](http://arduino.cc/en/Reference/Millis)()：返回时间函数（单位ms））该函数是指，当程序运行就开始计时并返回记录的参数，该参数溢出大概需要 50 天时间。

 [**delay**](http://arduino.cc/en/Reference/Delay)(ms)：延时函数（单位ms）

* [**delayMicroseconds**](http://arduino.cc/en/Reference/DelayMicroseconds)(us)：延时函数（单位us）。

**数学函数：**

 [**min**](http://arduino.cc/en/Reference/Min)(x, y) 求最小值

 [**max**](http://arduino.cc/en/Reference/Max)(x, y) 求最大值

 [**abs**](http://arduino.cc/en/Reference/Abs)(x) 计算绝对值

 [**constrain**](http://arduino.cc/en/Reference/Constrain)(x, a, b) 约束函数，下限a，上限b，x必须在ab之间 才能返回。

* [**map**](http://arduino.cc/en/Reference/Map)(value, fromLow, fromHigh, toLow, toHigh) 约束函数，

value必须在fromLow与toLow之间和fromHigh与toHigh之间。

* [**pow**](http://arduino.cc/en/Reference/Pow)(base, exponent) 开方函数，base的exponent次方。

 [**sq**](http://arduino.cc/en/Reference/Sq)(x) 平方

* [**sqrt**](http://arduino.cc/en/Reference/Sqrt)(x) 开根号 **三角函数：**
* [**sin**](http://arduino.cc/en/Reference/Sin)(rad)
* [**cos**](http://arduino.cc/en/Reference/Cos)(rad)
* [**tan**](http://arduino.cc/en/Reference/Tan)(rad)

**随机数函数：**

* [**randomSeed**](http://arduino.cc/en/Reference/RandomSeed)(seed) 随机数端口定义函数，seed表示读模拟 口analogRead(pin)函数 。

 long [**random**](http://arduino.cc/en/Reference/Random)(max) 随机数函数，返回数据大于等于 0，小于 max。

 long [**random**](http://arduino.cc/en/Reference/Random)(min, max) 随机数函数，返回数据大于等于

min，小于max。

**外部中断函数：**

* [**attachInterrupt**](http://arduino.cc/en/Reference/AttachInterrupt)(interrupt, , mode)：外部中断只能用到数字 IO口 2 和 3，interrupt表示中断口初始 0 或 1，表示一个功能函 数，mode：**LOW**低电平中断，**CHANGE**有变化就中断，**RISING** 上升沿中断，**FALLING** 下降沿中断。
* [**detachInterrupt**](http://arduino.cc/en/Reference/DetachInterrupt)(interrupt)：中断开关，interrupt=1 开，interrupt=0关。

**中断使能函数：**

* [**interrupts**](http://arduino.cc/en/Reference/Interrupts)() 使能中断
* [**noInterrupts**](http://arduino.cc/en/Reference/NoInterrupts)()终止中断

**串口收发函数：**

* [**Serial.begin**](http://arduino.cc/en/Serial/Begin)(speed) 串口定义波特率函数，speed表示波特率，如 9600，19200 等。
* **int** [**Serial.available**](http://arduino.cc/en/Serial/Available)() 判断缓冲器状态。

 **int** [**Serial.read**](http://arduino.cc/en/Serial/Read)() 读串口并返回收到参数。

* [**Serial.flush**](http://arduino.cc/en/Serial/Flush)() 清空缓冲器。
* [**Serial.print**](http://arduino.cc/en/Serial/Print)(data) 串口输出数据。
* [**Serial.println**](http://arduino.cc/en/Serial/Println)(data) 串口输出数据并带回车符

## 3.2、Arduino常用编辑语言

### 3.2.1 关键词

* [**if**](http://arduino.cc/en/Reference/If)
* [**if...else**](http://arduino.cc/en/Reference/Else)
* [**for**](http://arduino.cc/en/Reference/For)
* [**switch case**](http://arduino.cc/en/Reference/SwitchCase)
* [**while**](http://arduino.cc/en/Reference/While)
* [**do... while**](http://arduino.cc/en/Reference/DoWhile)
* [**break**](http://arduino.cc/en/Reference/Break)
* [**continue**](http://arduino.cc/en/Reference/Continue)
* [**return**](http://arduino.cc/en/Reference/Return)
* [**goto**](http://arduino.cc/en/Reference/Goto)

### 3.2.2语法符号

* [**;**](http://arduino.cc/en/Reference/SemiColon)
* [**{}**](http://arduino.cc/en/Reference/Braces)
* [**//**](http://arduino.cc/en/Reference/Comments)
* [**/\* \*/**](http://arduino.cc/en/Reference/Comments)

### 3.2.3运算符

* **=**
* **+**
* **-**
* **\***
* **/**
* **%**
* [**==**](http://arduino.cc/en/Reference/If)
* [**!=**](http://arduino.cc/en/Reference/If)
* [**<**](http://arduino.cc/en/Reference/If)
* [**>**](http://arduino.cc/en/Reference/If)
* [**<=**](http://arduino.cc/en/Reference/If)
* [**>=**](http://arduino.cc/en/Reference/If)
* [**&&**](http://arduino.cc/en/Reference/Boolean)
* [**||**](http://arduino.cc/en/Reference/Boolean)
* [**!**](http://arduino.cc/en/Reference/Boolean)
* [**++**](http://arduino.cc/en/Reference/Increment)
* [**--**](http://arduino.cc/en/Reference/Increment)
* [**+=**](http://arduino.cc/en/Reference/IncrementCompound)
* [**-=**](http://arduino.cc/en/Reference/IncrementCompound)
* [**\*=**](http://arduino.cc/en/Reference/IncrementCompound)
* [**/=**](http://arduino.cc/en/Reference/IncrementCompound)

### 3.2.4 数据类型

* [**boolean**](http://arduino.cc/en/Reference/BooleanVariables) 布尔类型
* [**char**](http://arduino.cc/en/Reference/Char)
* **Byte**
* [**int**](http://arduino.cc/en/Reference/Int)
* [**unsigned int**](http://arduino.cc/en/Reference/UnsignedInt)
* [**long**](http://arduino.cc/en/Reference/Long)
* [**unsigned long**](http://arduino.cc/en/Reference/UnsignedLong)
* [**float**](http://arduino.cc/en/Reference/Float)
* [**double**](http://arduino.cc/en/Reference/Double)
* [**string**](http://arduino.cc/en/Reference/String)
* [**array**](http://arduino.cc/en/Reference/Array)
* [**void**](http://arduino.cc/en/Reference/Void)

### 3.2.5数据类型转换

* [**char()**](http://arduino.cc/en/Reference/CharCast)
* [**byte()**](http://arduino.cc/en/Reference/ByteCast)
* [**int()**](http://arduino.cc/en/Reference/IntCast)
* [**long()**](http://arduino.cc/en/Reference/LongCast)
* [**float()**](http://arduino.cc/en/Reference/FloatCast)

### 3.2.6常量

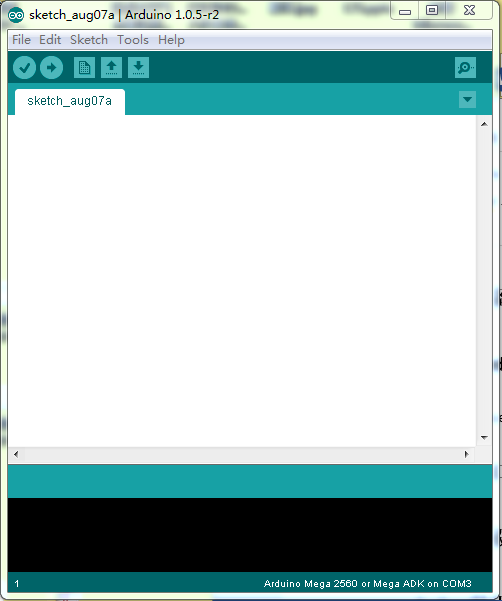
* [**HIGH**](http://arduino.cc/en/Reference/Constants) | [**LOW**](http://arduino.cc/en/Reference/Constants)表示数字IO口的电平，**HIGH**表示高电平（1），[**LOW**](http://arduino.cc/en/Reference/Constants) 表示低电平（0）
* [**INPUT**](http://arduino.cc/en/Reference/Constants) | [**OUTPUT**](http://arduino.cc/en/Reference/Constants) 表示数字IO口的方向，[**INPUT**](http://arduino.cc/en/Reference/Constants) 表示输入（高 阻态），[**OUTPUT**](http://arduino.cc/en/Reference/Constants)表示输出（AVR能提供 5V电压 40mA电 流）
* [**true**](http://arduino.cc/en/Reference/Constants) | [**false**](http://arduino.cc/en/Reference/Constants)[**true**](http://arduino.cc/en/Reference/Constants) 表示真（1），[**false**](http://arduino.cc/en/Reference/Constants)表示假（0）

# 四、Arduino实例

## 4.1、准备工作

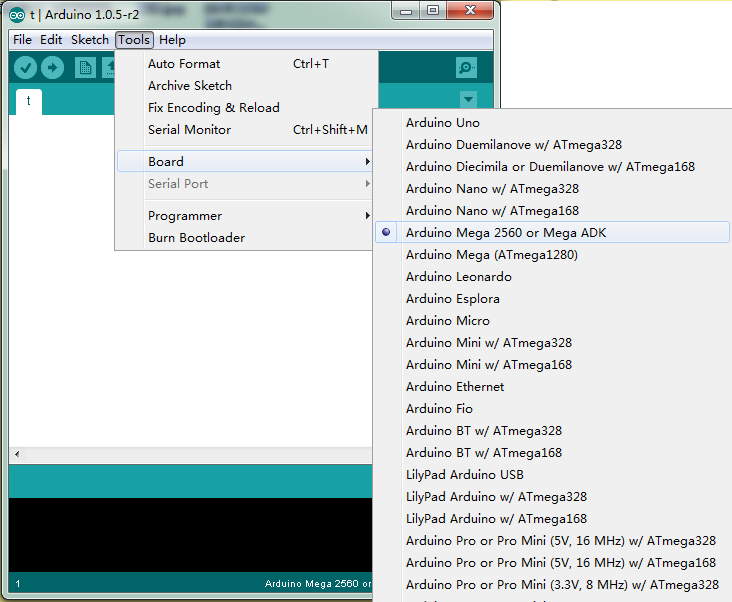
### 4.1.1、Arduino环境

（1）双击arduino.exe图标，打开Arduino IDE界面，如下图

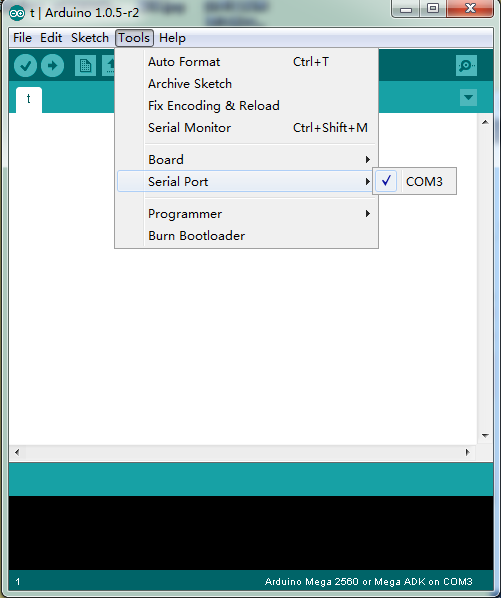


（2）设置板子型号与串口

通过Tools>Board>Arduino Mega 2560 or Mega ADK 设置好板子型号



通过Tools>Serial port 选择串口



设置好以后即可进行程序烧写。

## 4.2、串口监视器使用---“Hello World!”

Hello world程序：

void setup()

{

Serial.begin(115200);

}

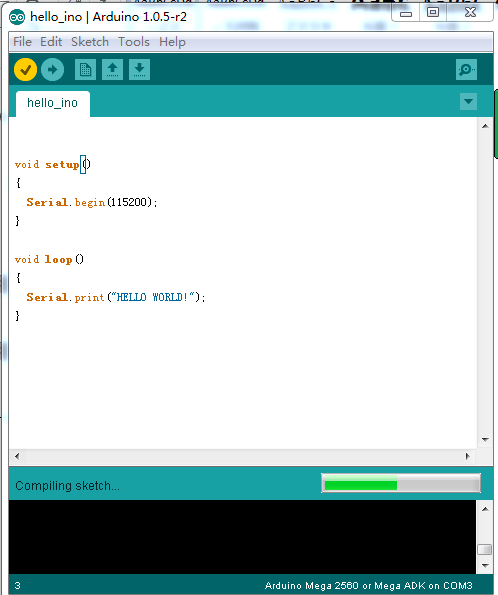
void loop()

{

Serial.print("HELLO WORLD!");

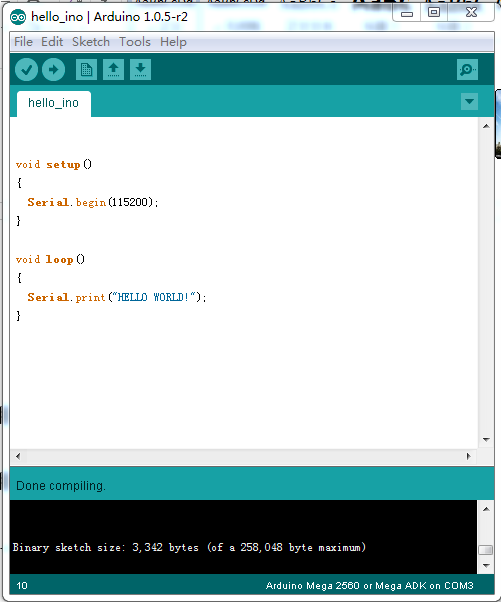
}

将程序输入后点击左上角的√，进行编译



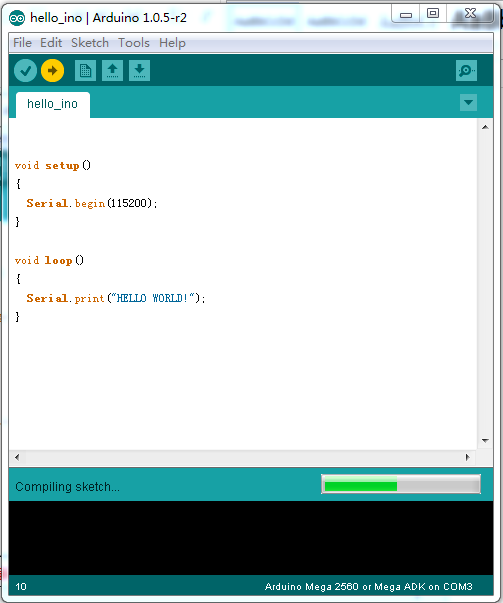
**显示正在编译**

**点击进行编译**



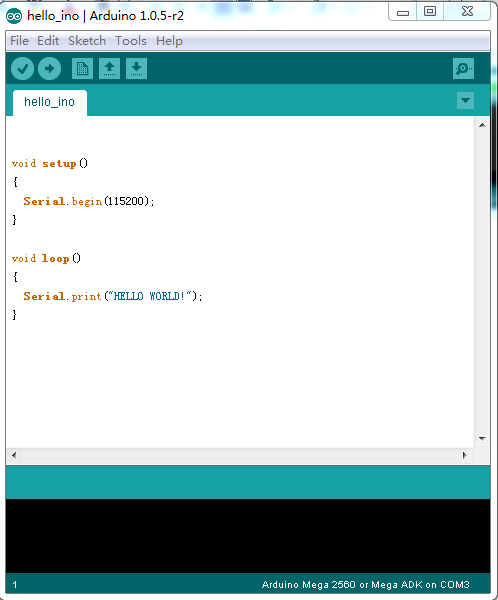
**编译完成**

将程序烧写到Arduino板子上：



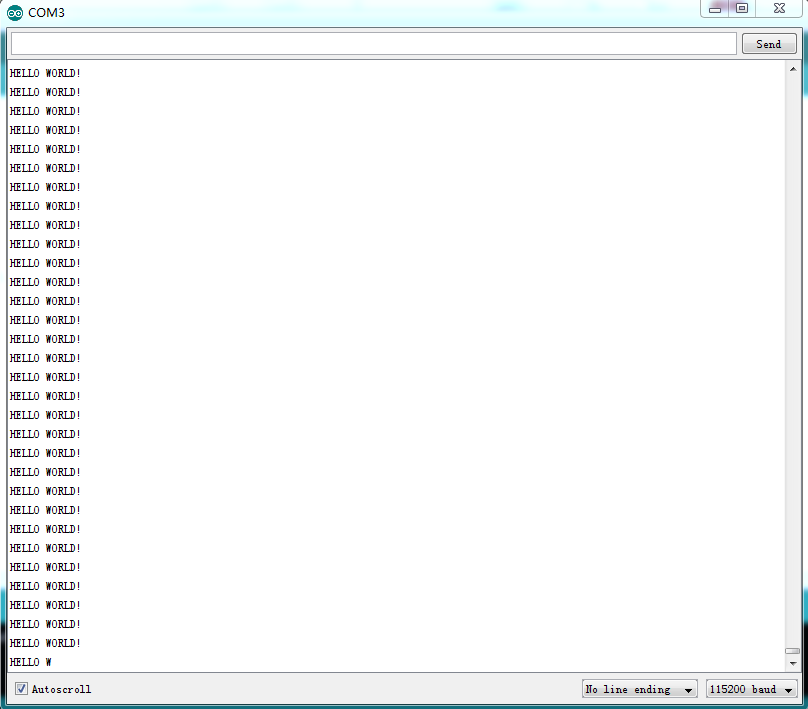
**点击进行烧写**

打开串口监视器，查看结果



**点击打开串口监视器**

打开串口监视器后可观察到如下现象：



注意将波特率按

Serial.begin(115200)率设置成115200

## 4.3、外设传感器使用---点亮普通LED

LED程序：

#define LED 1 //定义LED的管脚

void setup()

{

pinMode(LED,OUTPUT); //定义LED管脚的模式为OUPUT

}

void loop()

{

digitalWrite(LED,HIGH); //使LED管脚输出为高电平（HIGH）点亮LED

}

## 4.4、外设传感器使用---点亮双色LED

程序：

int redpin = 11; //给红色LED设置管脚

int bluepin =10; //给蓝色LED设置管脚

int greenpin =9; //给绿色LED设置管脚

int val; //定义变量

void setup()

{

pinMode(redpin, OUTPUT);

pinMode(bluepin, OUTPUT);

pinMode(greenpin, OUTPUT);

}

void loop()

{

for(val=255; val>0; val--)

{

analogWrite(11, val);

analogWrite(10, 255-val);

analogWrite(9, 128-val);

delay(1);

}

for(val=0; val<255; val++)

{

analogWrite(11, val);

analogWrite(10, 255-val);

analogWrite(9, 128-val);

delay(1);

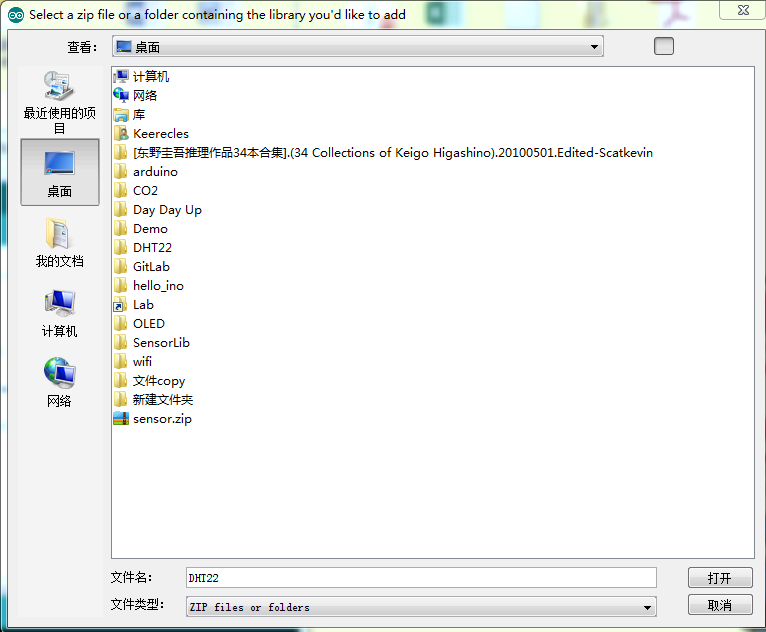
}

}

## 4.5、函数库的调用---DHT22温湿度传感器

向Arduino中加入需要的库文件：

点击 Sketch>Import Library >Add Library



选中文件夹后直接点击 “打开”即可。

程序：

#include <dht.h> //调用DHT22温湿度传感器函数库

dht DHT; //需要定义一个测量温湿度的对象 DHT

#define DHT22\_PIN 7

void setup()

{

Serial.begin(115200); //设置波特率

Serial.println("DHT TEST PROGRAM "); //打传感器信息

Serial.print("LIBRARY VERSION: ");

Serial.println(DHT\_LIB\_VERSION);

Serial.println();

Serial.println("Type,\tstatus,\tHumidity (%),\tTemperature (C)\tTime (us)");

}

void loop()

{

// READ DATA

Serial.print("DHT22, \t");

uint32\_t start = micros(); //计算程序运行时间

int chk = DHT.read22(DHT22\_PIN); //读取温湿度对象的返回值进行判断

uint32\_t stop = micros();

switch (chk)

{

case 0:

Serial.print("OK,\t");

break;

case 1:

Serial.print("Checksum error,\t");

break;

case 2:

Serial.print("Time out error,\t");

break;

default:

Serial.print("Unknown error,\t");

break;

}

// DISPLAY DATA

Serial.print(DHT.humidity, 1); //读取湿度 即DHT.humidity

Serial.print(",\t");

Serial.print(DHT.temperature, 1); //读取温度 即DHT.temperature

Serial.print(",\t");

Serial.print(stop - start);

Serial.println();

delay(2000);

}