



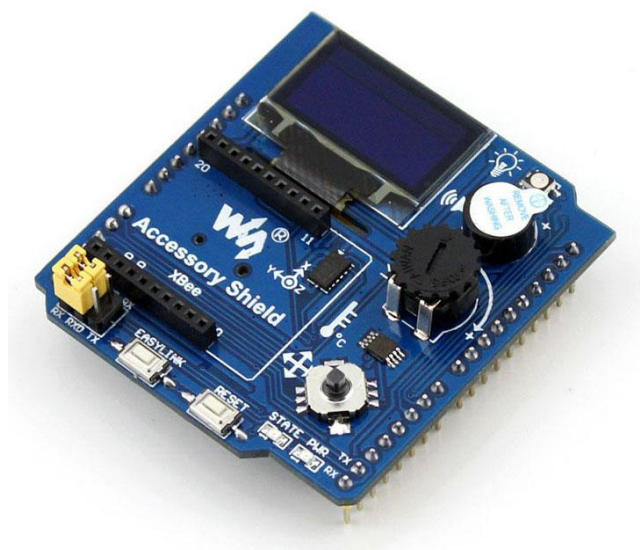
Accessory Shield 用户手册

产品概述

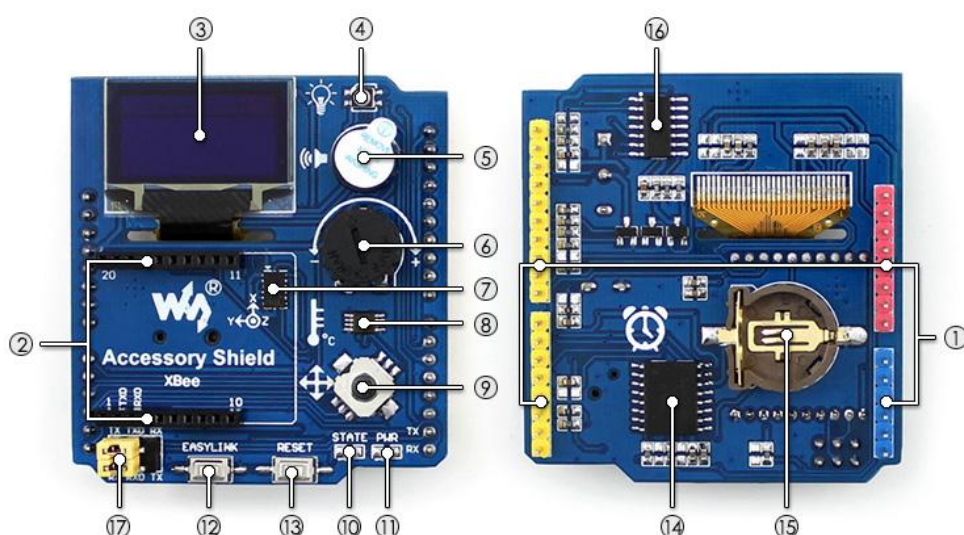
Accessory Shield 是微雪电子为 Arduino 开发的一款外围扩展板，集多种常用外围于一体。帮助用户缩短 Arduino 产品的学习和研发周期。

主要功能

- 五向摇杆
- 可调电位器
- 蜂鸣器
- 多彩 LED
- 温度传感器
- 三轴加速度计
- 高精度实时时钟
- 0.96 寸 OLED
- XBee 接口



1. 板载资源



[接口]

1. **Arduino 接口**
方便接入各种带 Arduino 接口的主板
2. **XBee 接口**
方便接入各种 XBee 通信模块

[器件]

3. **0.96inch OLED**
方便进行各种显示实验，分辨率 128×64
4. **RGB LED**
红绿蓝三基色 LED，可以显示 256×256×256 种颜色
5. **蜂鸣器**
PWM 连接，能演奏简单的音乐
6. **10K 可调电位器**
7. **ADXL345**
三轴加速度计，测量范围可达±16g
8. **LM75BDP**
温度传感器，分辨率可达 0.125℃

9. 五向摇杆

上、下、左、右和按下，共 5 种状态

10. **XBee 状态指示灯**
11. **电源指示灯**
12. **XBee EASYLINK 按键**
13. **Arduino 和 XBee 复位按键**
14. **DS3231**

高精度实时时钟

15. **CR1220 电池座**
实时时钟供电
16. **P9813**
RGB LED 驱动芯片

[跳线]

17. **调试/通信配置跳线**
当选择 TXD 与 TX，RXD 与 RX 相连时，可使用上位机软件通过 Arduino 主控板串口对 XBee 通信模块进行调试和配置
当选择 TXD 与 RX，RXD 与 TX 相连时，可通过 Arduino 主控板串口与 XBee 通信模块进行数据通信

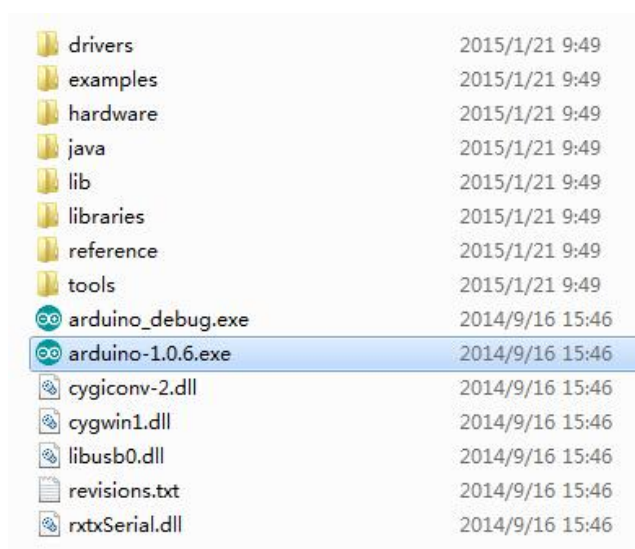
2. 产品使用

2.1 节介绍 Arduino IDE 的基本操作，包括用户自定义库的导入、程序编译和下载；2.2 节具体描述各个功能的操作方法和实验现象。文中所使用的开发板为 Arduino UNO PLUS，关于它的详细信息请参考附录。

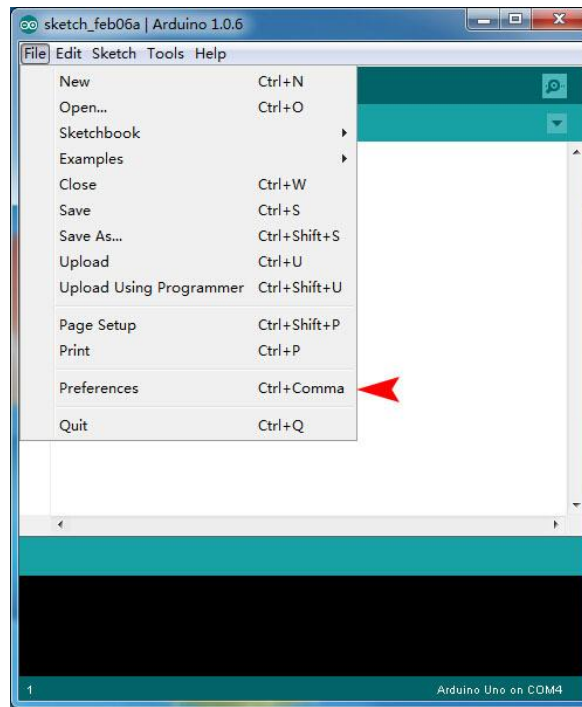
2.1. Arduino IDE 的使用

2.1.1. 库的导入

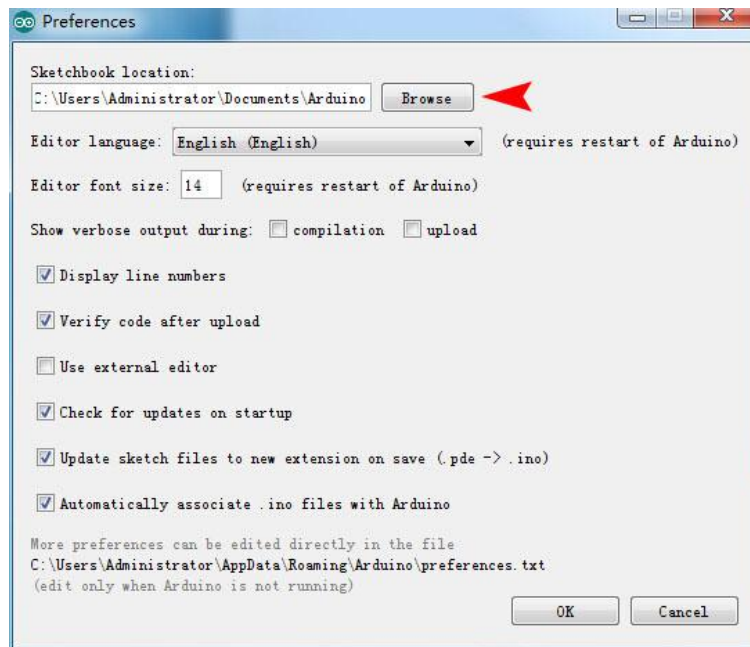
本手册以 Arduino 1.0.6 免安装版为例进行说明。打开 arduino-1.0.6.exe。



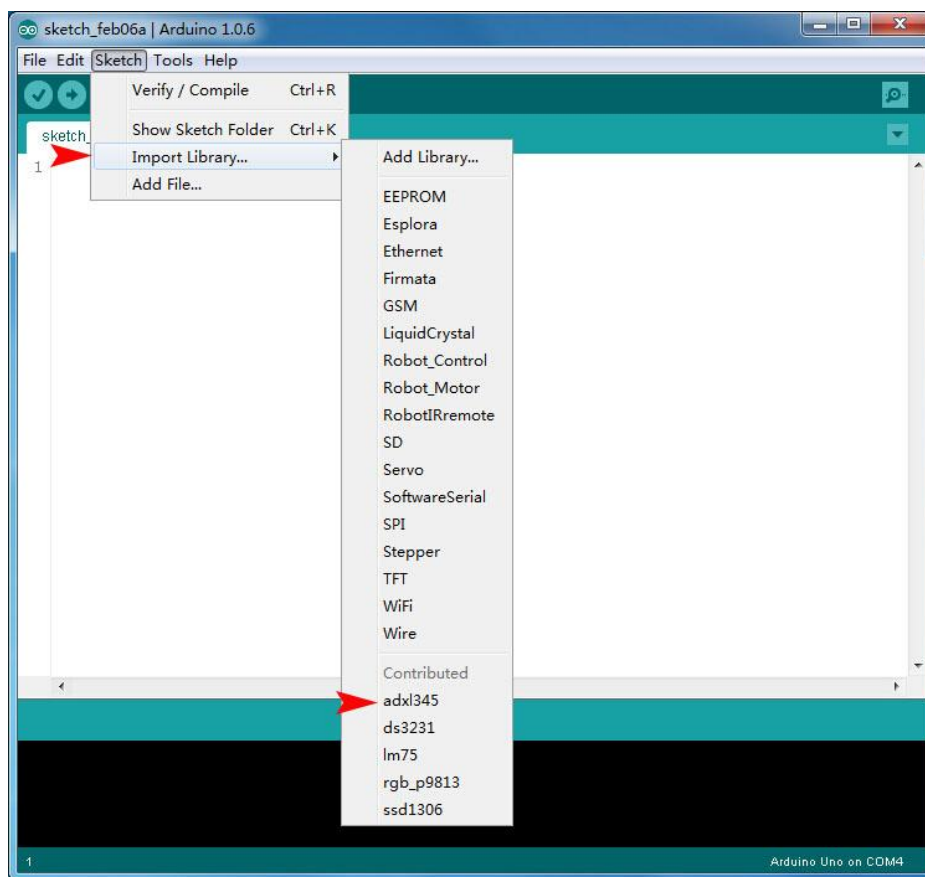
点击 File -> Preferences，进入配置界面：



将示例程序中的 libraries 文件夹复制到 Sketchbook location 所指向的文件夹内：

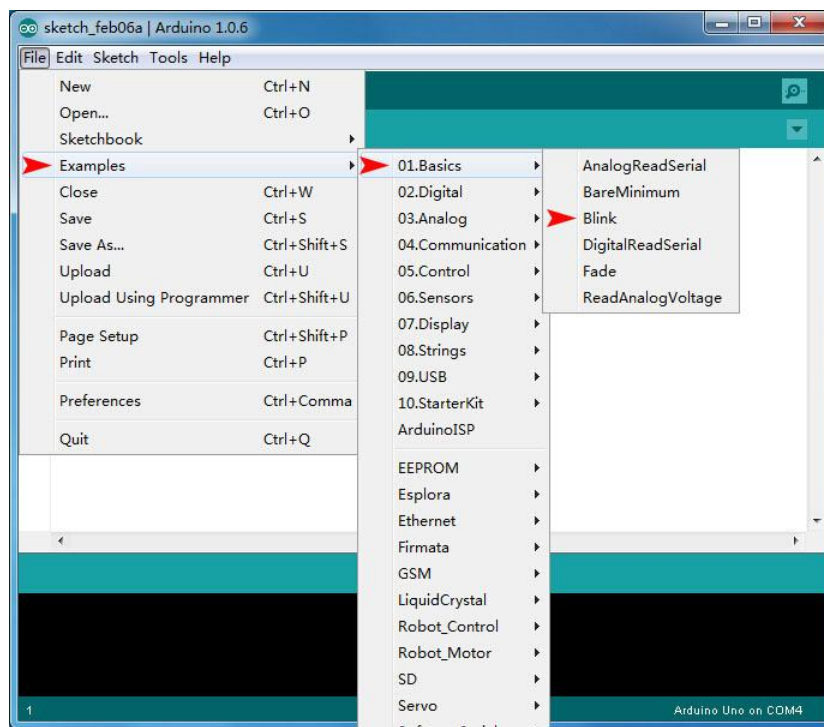


重启软件，单击 Sketch -> Import Library，查看是否有 adxl345、ds3231、lm75、rgb_p9813 和 ssd1306 选项。若有，说明库导入成功。

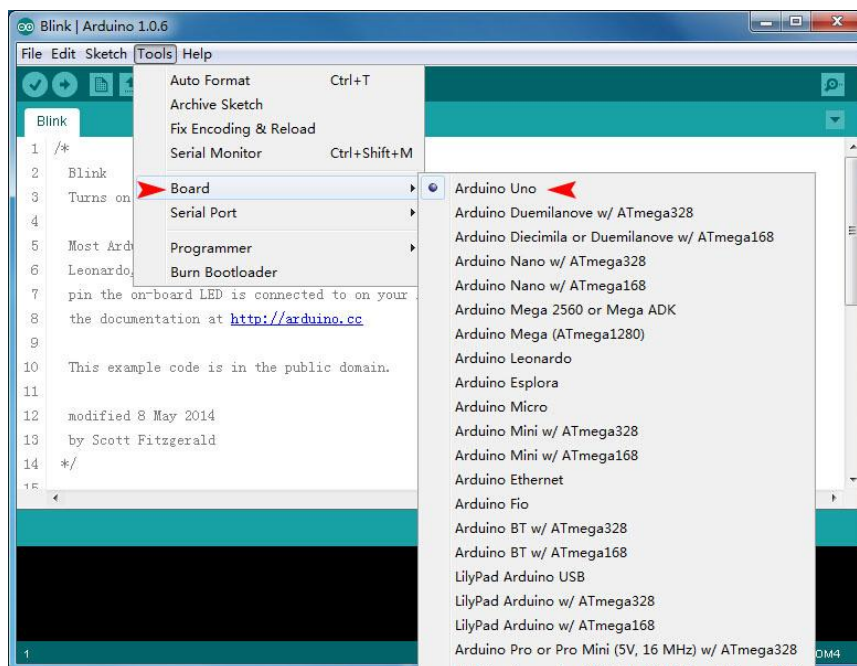


2.1.2. 程序编译和下载

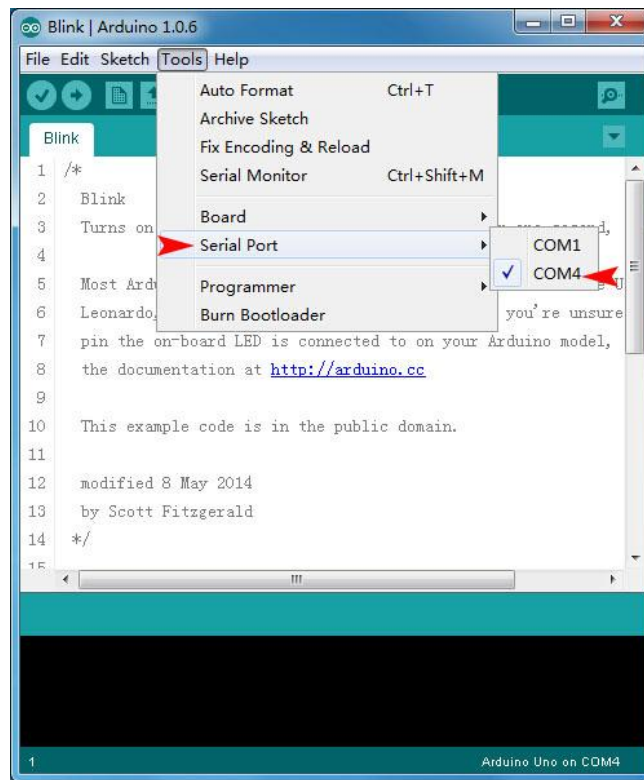
- 1) 将 UNO PLUS 开发板的 BOOT 模式设置到 ON 档，VCC 电压设置为 5V。
- 2) 接入 Accessory Shield 模块，通过 USB 线连接开发板并接入电脑。若模块电源指示灯点亮说明工作正常。
- 3) 本手册以 Arduino IDE 自带的示例程序来描述程序的编译及下载方法。依次点击 File -> Example -> 01.Basics -> Blink 即可打开示例程序。




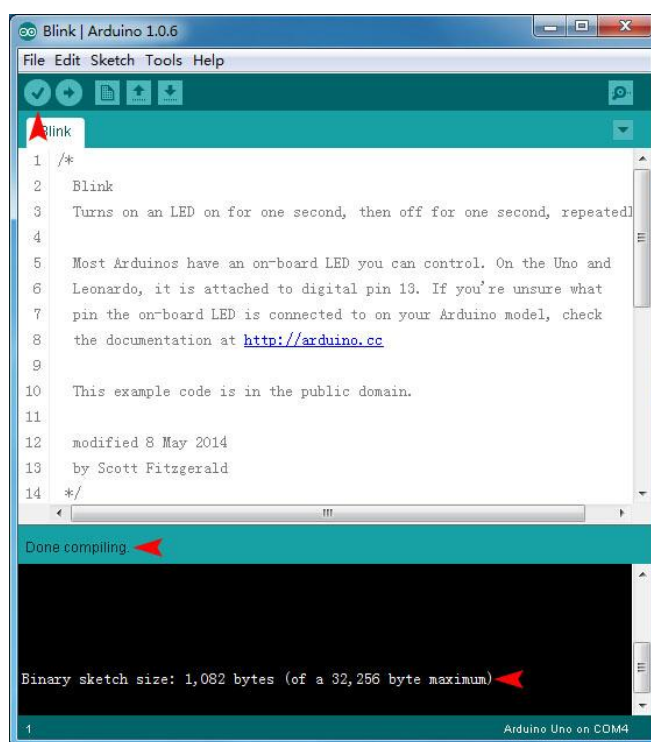
- 4) 点击 Tools -> Board -> Arduino Uno 选择 Arduino 开发板型号，这里只需要配置一次，之后操作都会默认使用该型号。




- 5) 点击 Tools -> Serial Port -> COM4 配置 Arduino 开发板的下载口，注意：这里的 COM4 会根据不同电脑而不同，需要通过设备管理器来查看具体的串口号。



- 6) 点击编译按钮 ，开始编译程序。没有错误的话，会提示“Done compiling”和生成的文件大小。



- 7) 点击下载按钮 ，开始下载程序。结束后会看到 UNO PLUS 板载 LED 以 1 秒周期开始闪烁，说明程序下载成功，运行正常。


2.2. 基础实验

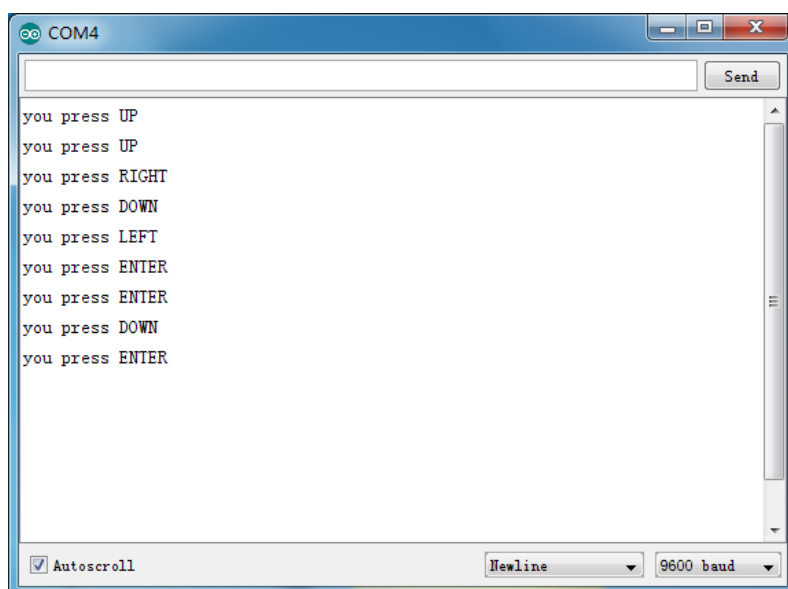
在进行基础实验之前，请确保例程中提供的库已经正确导入。

2.2.1. 五向摇杆

Joystick 5 个方向的操纵杆，包括上、下、左、右以及确定。每个按键分别与 Arduino 的引脚连接，通过检测 IO 口的高低电平来判断是哪个按键按下。

操作和现象：

编译并下载程序，单击 ，打开 Arduino IDE 自带的串口监视器。程序运行时板载的 LED 会以 1 秒为周期闪烁，提示系统正常工作；当按下任意一个按键时，串口监视器会显示当前按下的键位，例如 “you press UP”。现象见下图：




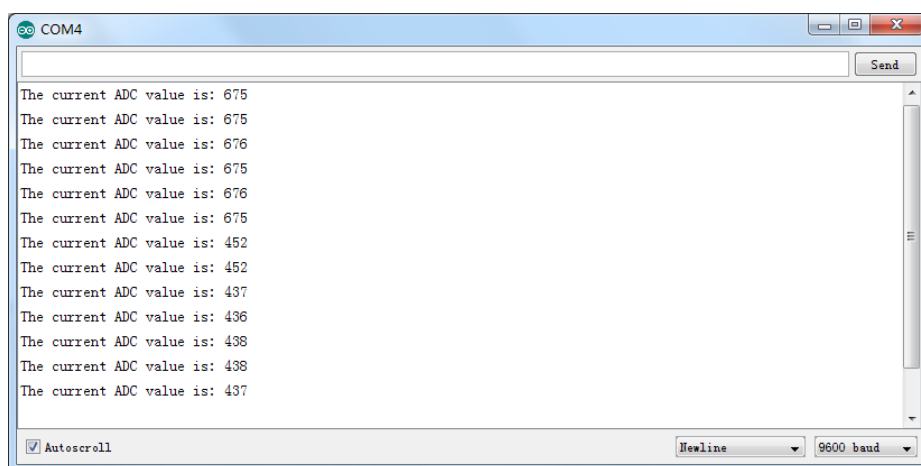
2.2.2. 可调电位器

UNO PLUS 开发板主芯片采用 ATMEGA328P-AU，该芯片供电电压范围广，适应 3.3V 及 5V 供电，内置 10 位 AD 转换模块，所以可提供 $2^{10}=1024$ 级模数转换。当系统的供电为 3.3V 时，对应的分辨率可以达到 $3300\text{mv} / 1024 = 3.22\text{mv}$ ，即每个 AD 值代表 3.22mv；若 5V 供电，分辨率为 $5000\text{mv} / 1024 = 4.88\text{mv}$ ，每个 AD 采样值代表 4.88mv。

Accessory Shield 板载的 10k 可调电位器采用 3.3V 供电，即最大可调节输出的电压值为 3.3V。若开发板供电电压配置为 3.3V，输出 AD 值的范围是 0 - 1023；若开发板供电电压配置为 5V，此时 ADC 的分辨率为 4.88mv，电位器最大输出电压 3.3V 对应的 AD 值为 $3300 / 4.88 = 676$ ，即输出 AD 值的范围是 0 - 676。

操作和现象：

编译并下载程序，单击 ，打开 Arduino IDE 自带的串口监视器。程序运行时可以看到当前的采样 AD 值，旋转电位器，输出值发生改变。现象见下图：



2.2.3. 蜂鸣器

有源蜂鸣器直接接上额定电源就可以发声，而无源蜂鸣器则和电磁扬声器一样，需要接在音频输出电路中才能发声。本模块使用的蜂鸣器为有源蜂鸣器，既可以直流驱动，发出一种单一频率的声音，也可以用可变频率的交流信号去驱动，发出简单的音乐声。

操作和现象：

编译并下载程序。可以听到蜂鸣器演奏简单的音乐。

2.2.4. 多彩 LED

多彩 LED 是通过 PWM 调制分别控制灯珠内部 R、G、B 三个 LED 晶片以显示不同的亮度，三原色混合就可以显示丰富的色彩。

操作和现象：

编译并下载程序。可以看到 RGB 实现三种颜色的渐变显示。

2.2.5. 温度传感器：LM75B

LM75B 是一款支持 I2C 接口的测温芯片，应用电路简洁，仅仅需要一颗去耦电容就可以工作。另外它具有 3 个可配置逻辑地址的管脚，使得同一总线上可同时连接 8 个器件而不发生地址冲突。

芯片特性如下：

- 器件完全可以取代工业标准的 LM75 和 LM75A
- I2C 总线接口，同一总线上可连接多达 8 个器件，总线速度可达 400KHz，具有总线失效功能，总线失效自动恢复，防止锁死总线
- 电源电压范围：2.8V~5.5V
- 环境温度范围：-55℃~125℃
- 测量分辨率：0.125℃
- 测量精度：
 - 25℃~100℃：误差 2℃
 - 55℃~125℃：误差 3℃
- 超低功耗，关断模式下电流仅为 1.0μA
- ESD 保护


LM75 作为从机地址是 7 位的。

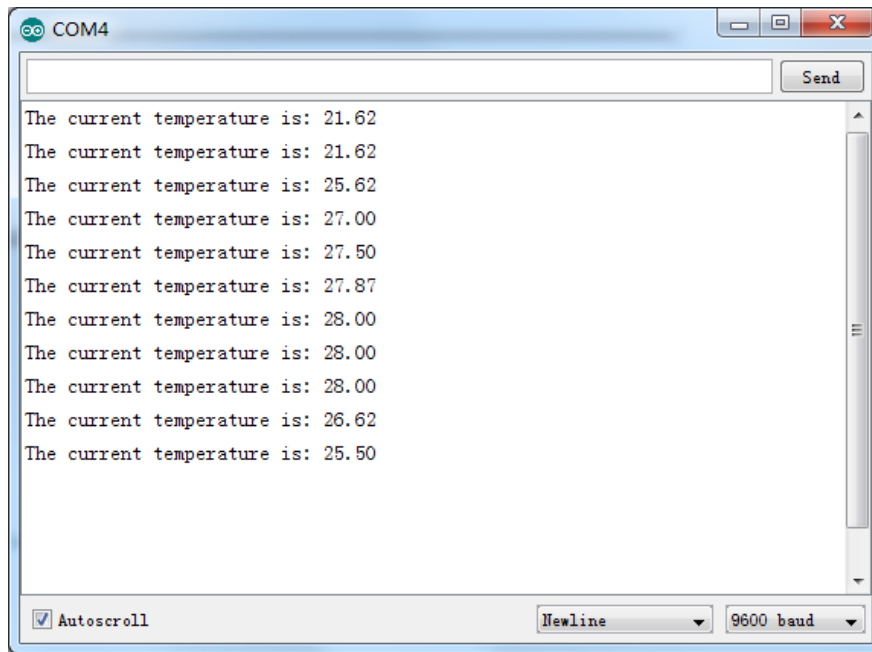
1=高电平，0=低电平

MSB				LSB		
1	0	0	1	A2	A1	A0

模块中 A2/A1/A0 引脚全部短接到地，所以从机地址就是 0x48。

操作和现象：

编译并下载程序，单击 ，打开 Arduino IDE 自带的串口监视器。会显示当前的温度信息，用手触摸 LM75 芯片，温度会发生变化，具体现象如下图：




2.2.6. 三轴加速度计：ADXL345

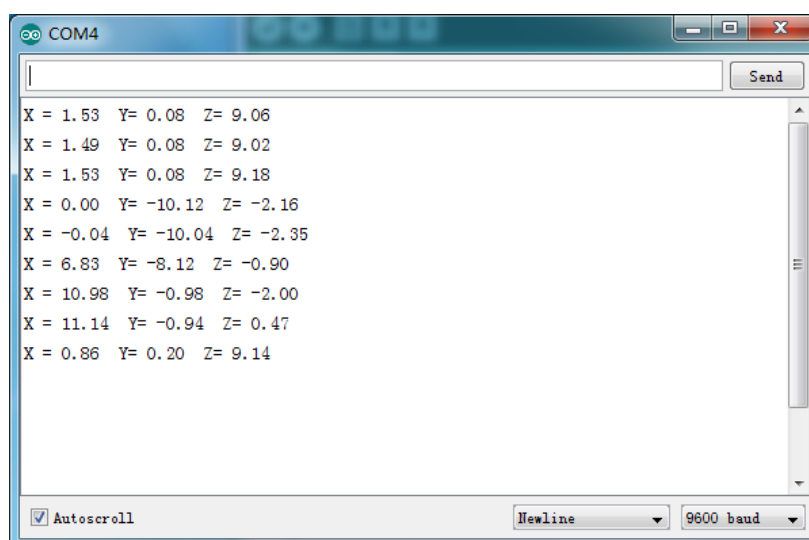
ADXL345 是 ADI 公司推出的一款小而薄的超低功耗 3 轴加速度计，13bit 分辨率可达 4 mg/LSB，测量范围达 $\pm 16g$ 。数字输出数据为 16 位二进制补码格式，可通过 SPI（3 线或 4 线）或 I2C 数字接口访问。ADXL345 非常适合移动设备应用。它可以在倾斜检测应用中测量静态重力加速度，还可以测量运动或冲击导致的动态加速度。其高分辨率（3.9mg/LSB），能够测量不到 1.0° 的倾斜角度变化。

芯片特性如下：

- 电源电压范围：2.0V 至 3.6V
- I/O 电压范围：1.7V 至 VCC
- 温度范围： -40°C 至 $+85^\circ\text{C}$
- 尺寸：3mm x 5mm x 1mm，LGA 封装

操作和现象：

编译并下载程序，单击 ，打开 Arduino IDE 自带的串口监视器。会显示当前开发板三轴位置的加速度大小，旋转开发板，输出值发生变化，具体现象如下图：




2.2.7. 高精度实时时钟：DS3231

DS3231 是低成本、高精度 I2C 实时时钟（RTC），具有集成的温补晶体振荡器（TCXO）和晶体。该器件包含电池输入端，断开主电源时仍可保持精确的计时。集成的晶体振荡器大大提高了器件的长期精确度。该时钟可以记录秒、分、时、星期、日期、月和年信息，少于 31 天的月份，将自动调整月末日期，包括闰年补偿。

芯片特性如下：

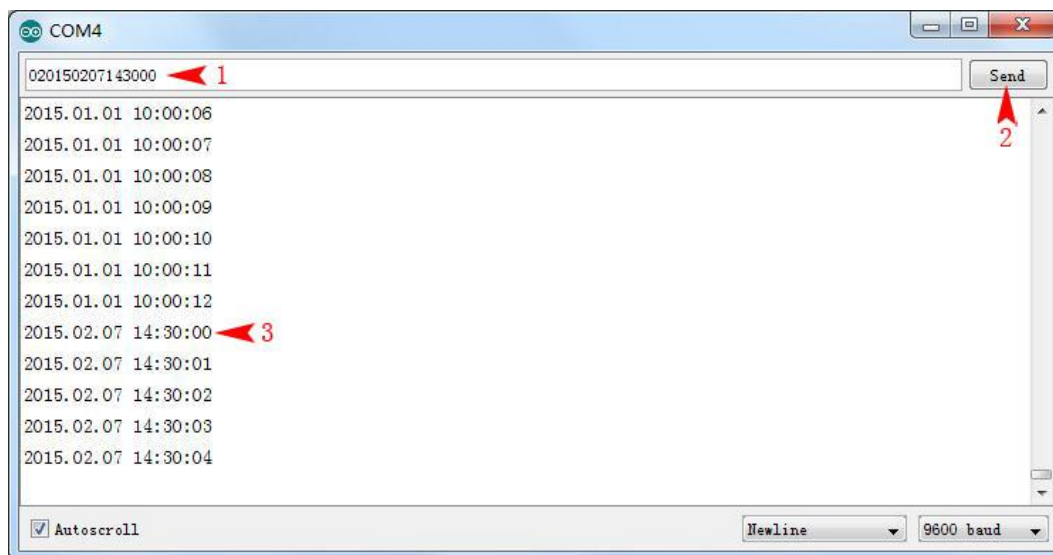
- 工作电压：2.3V 至 5.5V
- 工作温度范围
商用级：0°C 至 +70°C
工业级：-40°C 至 +85°C
- 0°C 至 +40°C 精度为 ±2ppm
- -40°C 至 +85°C 精度为 ±3.5ppm

操作和现象：

编译并下载程序，单击 ，打开 Arduino IDE 自带的串口监视器。本示例程序提供两个功能：通过串口输出当前的时间；可以通过串口设置新的时间。

如下图：

1 号箭头位置是待设置的新时间：020150207143000，第 1 个字符为 0，后面跟年(2015)、月(02)、日(07)、时(14)、分(30)、秒(00)共 15 个字符，输入完毕后，点击 2 号箭头发送，会看到新的时间已经写入。**注意：时间格式为 24 小时制；数据长度必须为 15 个字符，例如 2 月需要写成 02，其他数据段写法类似。**



2.2.8. 0.96 寸 OLED：SSD1306

Accessory Shield 配备了一款 0.96inch，128 x 64 分辨率大小的 OLED 显示屏，特性如下：

- 驱动芯片：SSD1306
- 接口：I²C
- 颜色：黄蓝双色块
- 视角：>160°
- 工作温度：-20°C 至 70°C
- 存储温度：-30°C 至 80°C

操作和现象：

编译并下载程序，示例程序展示了基本的绘图操作，包括画线、画圆、矩形框等。

2.2.9. XBee 接口

XBee 是美国 MaxStream 公司基于 ZigBee 技术开发的一款无线传输模块，该模块使用简单，只需通过串口进行数据的收发，并且支持利用 AT 命令进行高级配置。

实验准备:

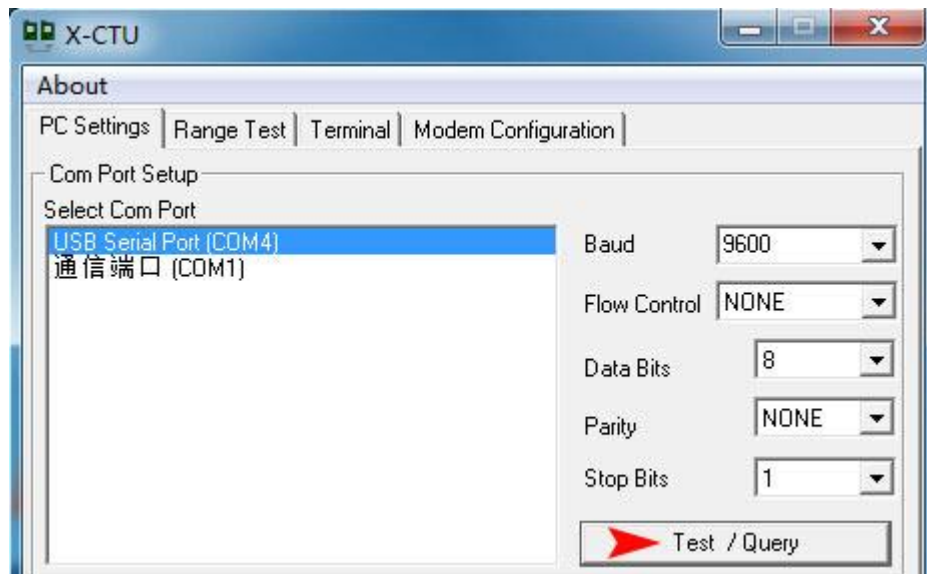
两个 XBee 模块、两个 Accessory Shield 以及两个 Arduino UNO PLUS。为方便阅读，本手册将这两组称为 A 和 B。

操作和现象:

- 1) 分别把 XBee 模块接入 Accessory Shield 的 XBee 接口，Accessory Shield 接入 Arduino 开发板，通过 USB 口与 PC 连接。
- 2) 设置 Accessory Shield 跳线：
 - TXD 连接 TX
 - RXD 连接 RX

注意：下面步骤会用到 Arduino 的 USB 接口和 TX、RX 管脚，请确保当前 Arduino 上运行的程序没有对串口的操作。如果有，可以重新下载一个新的程序，例如示例中的蜂鸣器例程。

- 3) 电脑端运行 X-CTU 软件配置 XBee 模块，界面如下：

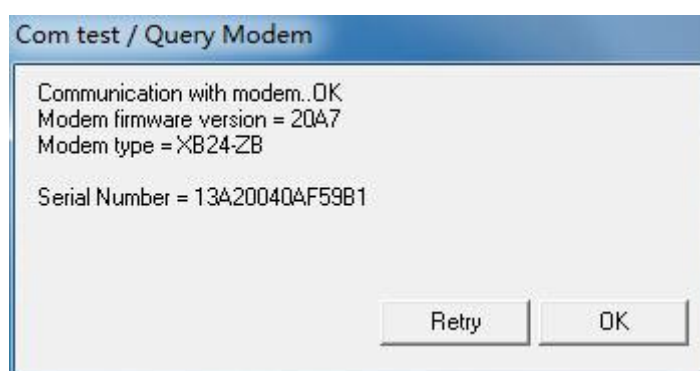


- 4) PC Settings 选项卡中选择对应的串口，并设置好波特率等参数，XBee 模块出厂默认设置为：
 - Baud: 9600

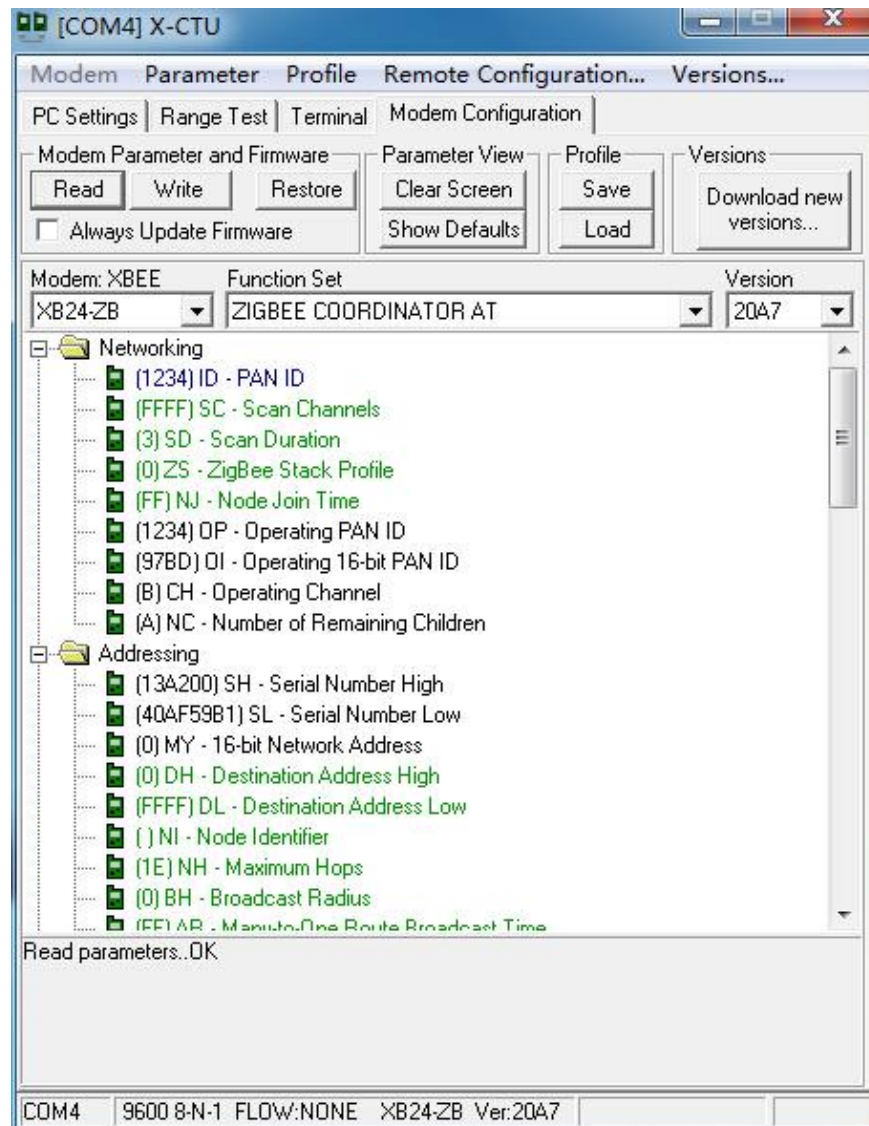
- Data Bits: 8
- Flow Control: NONE
- Parity: NONE
- Stop Bits: 1

Xbee 模块在正常工作之前必须进行相应的设置,对 Xbee 模块的设置除了可以用 X-CTU 这一工具完成之外,还可以按照 Xbee 手册里介绍的 AT 指令,通过串口终端完成。为方便使用,本手册只介绍 X-CTU 工具。

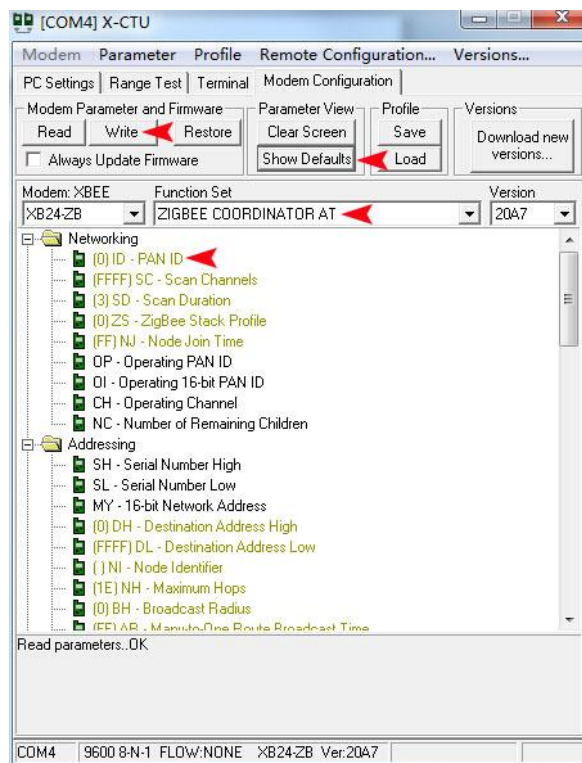
- 5) 单击 Test/Query, 如果一切正常, 出现如下对话框。若通信参数设置正确, 仍无法与 Xbee 模块通信, 请检查 USB 连线和 Accessory Shield 上的跳线。



- 6) 转到 Modem Configuration 选项卡, 点击 Read, 读取 Xbee 模块当前参数:

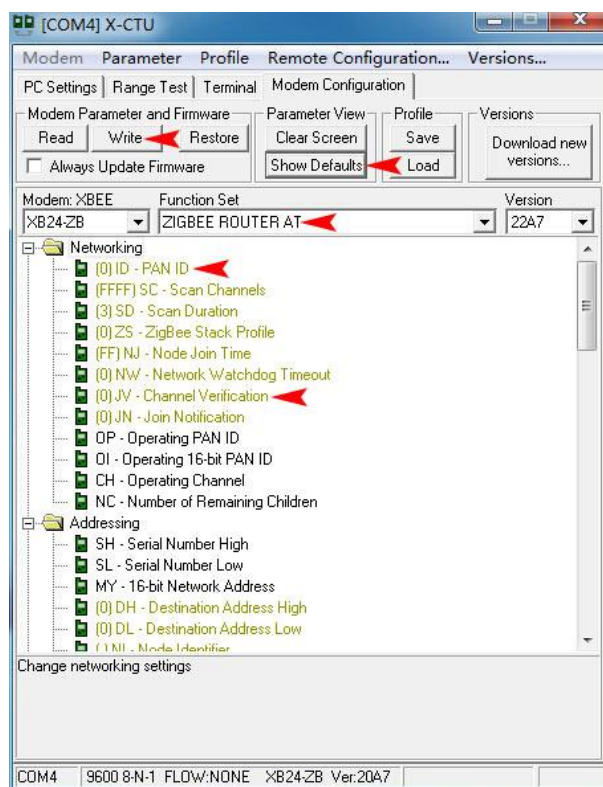


- 配置 XBee 模块 A

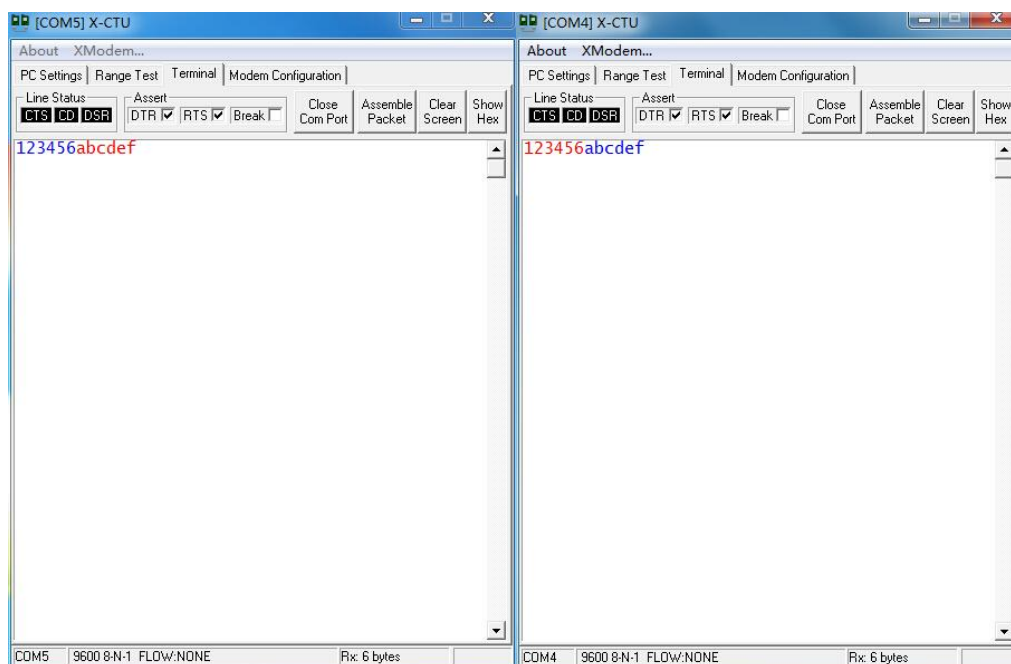


- a) 在 Function Set 下选择 ZIGBEE COORDINATOR AT
- b) 单击 Show Defaults
- c) 选择 PAN ID，修改为 1234
- d) 点击 Write，A 配置完成

- 配置 XBee 模块 B



- a) 在 Function Set 下选择 ZIGBEE ROUTER AT
 - b) 单击 Show Defaults
 - c) 选择 PAN ID, 修改为 1234
 - d) 选择 Channel Verification(JV), 修改为 1-ENABLED
 - e) 点击 Write, B 配置完成
- 7) 请按照以上方式配置好 XBee-A 和 XBee-B, 运行两个 X-CTU, 并在 PC Settings 选型卡中选择不同的通信接口, 分别对 A 组和 B 组进行控制。
- 8) 在 X-CTU 的 Terminal 中, 输入待发送的数据后, 会自动发送到 XBee-B 模块, 并在另一个 X-CTU 的 Terminal 中显示出来。其中蓝色表示发送的数据, 红色表示接收到的数据。



- 9) 如果运行状态如上图所示，则 XBee 模块能够正常的收发数据。用户可以在 Arduino 工程里利用 XBee 进行无线数据的收发，只需将要发送的数据通过 Arduino 串口发送给 XBee 模块，然后在另一个 Arduino 模块中通过串口读出数据即可。

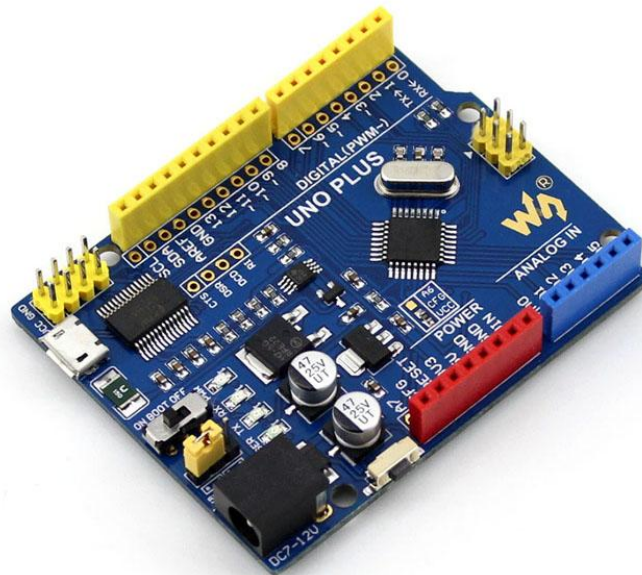
注意，此时需要设置 Accessory Shield 跳线：

- TXD 连接 RX
- RXD 连接 TX

3. 附录：

3.1. Arduino UNO PLUS

UNO PLUS 是兼容 Arduino UNO R3 的开发板，基于 ATMEGA328P-AU，改进了 UNO R3 的部分缺陷。



关于本开发板的详细信息可参考下面链接：

中文：<http://www.waveshare.net/shop/UNO-PLUS.htm>

英文：<http://www.waveshare.com/product/UNO-PLUS.htm>