计算机不同软件与Arduino的互联

——使用Processing，Mathematica，Unity3D等

初稿由毛勇老师撰写

*本文档中相关网址可参考项目reference文件夹，源代码可参考项目src文件夹*

Arduino可以为开发者提供接口，连接各类数据采集传感器（sensor），将现实世界中的模拟信号，传递到计算机。同时，计算机向Arduino发出指令，也可以通过操作致动器（actuator），对现实世界中的事物进行操作。

计算机对Arduino的通信控制可以通过串口来实现，这也是最常用的方法。此外，也可以通过蓝牙、无线网等方式通信，此处暂不讨论。对于串口通信，Arduino端通过USB线连接，而计算机端通过USB连接后虚拟为一个串口，而后可以有不同的方式，让软件访问这个串口的数据，最终与Arduino通信。

1. 直接访问串口

Arduino IDE、Processing、Mathematica等均可以通过加载串口通讯相关的库（library）或包（package）来直接访问串口。

1. 通过代理将串口与网络端口对应

由于串口通信一次只允许一个对象（client）访问，不支持多个程序并行访问，且对于浏览器、Flash等程序，不支持直接访问串口，另一种办法则是通过特定程序（如Serproxy）将串口与本地计算机的一个网络端口对应起来，这样就可以由多个程序，通过本机的同一个网络端口，与Arduino进行通信。如图所示。

计算机

Arduino

温度

串口通信（serial port）

Arduino IDE

Serproxy

Flash

Mathematica

Unity3D

传感器

致动器

虚拟网络通信

（Serial-to-Network）

Processing

光线

……

电动机

扬声器

……

串口

可通过串口

直接通讯，

也可利用网

络端口进行

并行通讯。

1. *什么是Processing、Mathematica、Unity3D和Arduino*
   1. *Processing简介*

*参见：*

[*http://en.wikipedia.org/wiki/Processing\_(programming\_language*](http://en.wikipedia.org/wiki/Processing_(programming_language)*)*

[*http://www.processing.org*](http://www.processing.org)

* 1. *Mathematica简介*

*参见：*

[*http://en.wikipedia.org/wiki/Mathematica*](http://en.wikipedia.org/wiki/Mathematica)

* 1. *Unity3D简介*

*参见：*

[*http://unity3d.com/*](http://unity3d.com/)

* 1. *Arduino简介*

*参见：*

[*http://en.wikipedia.org/wiki/Arduino*](http://en.wikipedia.org/wiki/Arduino)

[*http://www.arduino.cc*](http://www.arduino.cc)

1. Processing与Arduino互动

Processing与Arduino可以很方便地互相连接起来，从而实现虚拟世界(Processing)与物理实体世界(Arduino)之间的互动。

* 1. 互动原理

Processing与Arduino之间的互动是依靠TTL串口(Serial port)实现的，TTL串口是实现计算机之间进行通信的一个重要接口标准。通过它可以以不同的通信速率（波特率）在两台计算机之间实现全双工连接。

Processing与Arduino都可以通过串口发送和接收数据，这就为它们之间的互动提供了物理原理上的可能。显然，除了串口的通信标准外，我们需要对通信双方进行正确地设置（或通过编程）才可能正确解析数据，包括：

* 采用同样的波特率
* 采用一定的数据通信协议

本文档介绍Processing与Arduino通信的2种方法，用户可根据自身需要选择使用。

* 1. 通过烧录到Arduino上的固件及Processing中的Arduino库，来实现单向控制和数据采集

如果用户只需要用Processing控制Arduino，即只实现Processing到Arduino的主动控制和数据采集，而不需要Arduino的数据处理和到Processing的主动连接，则可以使用本小节的方法。

1. 按照标准流程安装Arduino IDE，并将计算机与arduino连通，安装好驱动程序。具体方法参见：<http://arduino.cc/en/Guide/HomePage>

当遇到驱动安装问题时，可参考如下步骤：

<http://arduino.cc/en/Guide/ArduinoLeonardoMicro?from=Guide.ArduinoLeonardo>

1. 下载Processing 2的Arduino库（library），并安装到Processing的Sketchbook文件夹中。从而在Processing中获得对Arduino的支持。具体方法参见<http://playground.arduino.cc/Interfacing/Processing>
2. 在Arduino IDE中，将Firmata固件烧录到Arduino板中。具体方法为打开File菜单，选择Examples – Firmata – StandardFirmata打开固件程序，再上传（upload）到Arduino中。
3. 此时，打开Processing，通过import processing.serial.\*;导入相关库函数，即可调用Arduino中的诸多实用函数进行编程并测试。
   1. 通过自定义协议的双向连接\*

如果高级用户期望更灵活地实现Processing和Arduino的交互，则可以通过自定义通信协议的方式进行连接。此时，用户需要同时为Processing和Arduino编写程序，它们均调用基本的串口通信函数发送和接收二进制数据或ASCII字符串，而通信的建立（如简单的握手协议的编写）和数据的解读均由用户程序完成。

具体示例请参见：

<http://arduino.cc/en/Tutorial/VirtualColorMixer>

<http://arduino.cc/en/Tutorial/SerialCallResponse>

<http://arduino.cc/en/Tutorial/SerialCallResponseASCII>

1. Mathematica与Arduino互动
   1. 互动原理

同Processing与Arduino的互连相同，Mathematica与Arduino同样通过串口实现连接通信。下面介绍2种Mathematica中访问串口的方法：

* 使用SerialIO
* 使用.NET/Link
  1. 使用SerialIO

SerialIO是由第三方开发的，为Mathematica提供串口读写能力的包(Addon)。它不仅支持Windows操作系统，也支持Linux和Mac。

请参见：

<http://library.wolfram.com/infocenter/MathSource/5726/>

值得注意的是，如果需要在Mac下使用SerialIO，则必须进行一些额外的设置<http://williamjturkel.net/2011/12/25/connecting-arduino-to-mathematica-on-mac-os-x-with-serialio/>

* 1. 使用.NET/Link

如果用户使用的是Windows操作系统，并安装了微软.NET framework 2.0以上版本。则可以立即使用.NET/Link从Mathematica使用串口进行读写。

对Mathematica中.NET/Link的使用请参见：

<http://playground.arduino.cc/Interfacing/Mathematica>

* 1. 与Arduino的连接

成功令Mathematica具备访问串口的能力后，用户马上可以采用2.3中同样的原理和方法，通过自定义串口通信协议并编程解读串口数据实现二者互连。

1. Processing与Mathematica互动
2. 使用Serproxy代理网络端口与Arduino通信

当我们需要通过多个程序读取或写入串口时，为了避免串口读写冲突，可以先利用代理程序，将串口数据对应到一个代理网络端口。这样还可以让一些无法访问串口的程序，例如Flash，可以很容易地通过代理网络端口与Arduino通信。

* 1. 安装配置Serproxy

下载Serproxy程序（<http://www.lspace.nildram.co.uk/freeware.html>）

打开serproxy.cfg文件，进行配置。配置方法参考serproxy提供的README文档。（可以将time out的值设置的高一些，避免调试时因空闲而导致serproxy自动关闭）

配置完毕后，记下与Arduino串口对应的网络端口。

* 1. 测试

打开命令行（Windows在开始菜单-运行中输入cmd，MacOS打开terminal），输入ipconfig查看本机ip地址

打开Serproxy，显示

Serproxy - (C)1999 Stefano Busti - Waiting for clients

打开浏览器，输入“本机ip地址:串口对应的端口号”，如166.111.59.15:5334

在Serproxy上如果出现提示：

Failed to open comm port - connection refused

Failed to open comm port - connection refused

Failed to open comm port - connection refused

表明网络端口对应的串口没有连接。

如果出现提示：

Server thread launched

server(4) - thread started

server(4) - EOF from sock

server(4) exiting

Server thread launched

server(4) - thread started

说明端口已经连通，可以在代理客户（client）程序中通过网络端口访问串口

1. Processing与网络端口连接
2. Mathematica与网络端口连接
3. Unity3D与网络端口连接
4. Flash与网络端口连接