C#程序设计语言与

.Net框架基础（双语）

**实**

**验**

**报**

**告**

**学 院 软件学院**

**年 级 2017 级**

**班 级 4 班**

**学 号 3017218178**

**姓 名 张剑锋**

**2019 年 4 月 18 日**



**C#程序设计语言与.Net框架基础（双语）**

**上机实验报告**

实验4：C# .NET 综合应用程序开发

**学院名称 软件学院**

**专 业 软件工程**

**学生姓名 张剑锋**

**学 号 3017218178**

**年 级 2017 级**

**班 级 4 班**

**时 间 2019年 4月18日**

目 录

[实验名称 1](#_Toc509419319)

[实验目的 1](#_Toc509419320)

[实验内容 1](#_Toc509419321)

[实验要求 1](#_Toc509419322)

[实验分析 1](#_Toc509419323)

[实验结论及心得体会 6](#_Toc509419324)

## 实验名称

C# .NET 综合应用程序开发

## 实验目的

1. 理解和掌握WPF的图形界面APP开发方法：WPF、数据binding、控件的事件机制。
2. 学会使用.NET平台的串口类SerialPort，借助MIDI通信协议，实现PC与Arduino装置间进行信息交换（数据的接收/发送）。
3. 掌握C#数据转换方式和GUI界面的数据显示/更新方法。
4. 不同数据类型在RAM中的存储方式（little endian/ big endian），能实现数据的MIDI通信协议的编码/解码。
5. 使用Arduino编程实现Arduino MIDI控制器，在PC端实现MIDI演奏功能。

## 实验内容

1. 完成WPF的APP应用界面，使用相对布局方式，界面要基本保证用户体验良好、外观整洁、布局合理匀称。此处布局仅供参考，同学需改善界面的用户体验。
2. 能够依据MIDI通信协议的数据传输定义（参照机组原理的实验3定义的传输格式），APP能对MIDI格式的数据进行接收与发送，并在PC端实时显示其16进制通信数值、物理值和是实时图形曲线。
3. 能够实现通过MIDI协议，控制Arduino板上PWM输出端，对LED明暗进行控制。
4. 能够以csv格式或json文件格式记录MIDI数据的结果。
5. 实现有创意的Arduino-MIDI控制器程序

## 实验要求

1. 以C# WPF的方式编写本实验的程序；
2. 界面需要使用相对布局方式，且需要保证用户体验良好、外观整洁、布局合理均匀；
3. 使用Midi传输协议的方式传输数据；
4. APP能对MIDI格式进行接受和发送，并能在PC端显示通信数值、物理值和实时图形曲线；
5. 能够记录MIDI数据的结果

## 实验分析

1. 实验结果

1.1项目名称：

GUI界面布局的设计

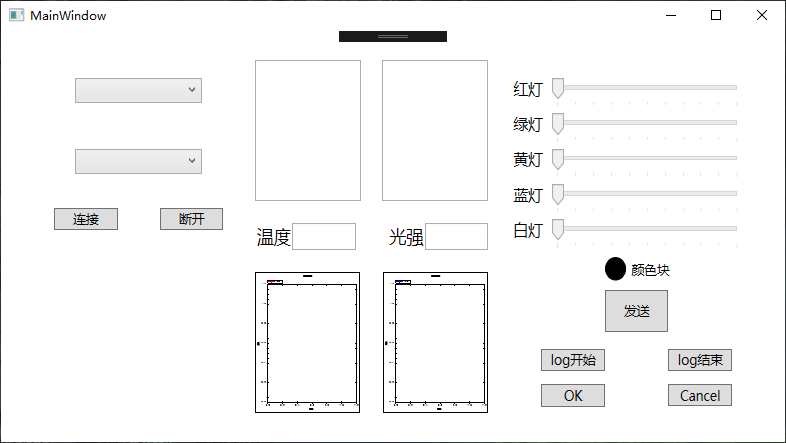
1.2操作步骤：

使用WPF控件完成界面的设计，并通过一个具有Stretch=“Fill”属性的ViewBox使得界面上的所有控件的大小会随着主窗口的拉伸而拉伸，且保持相对位置不变；

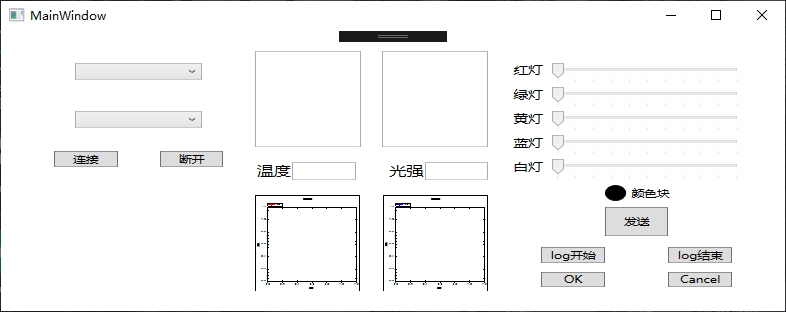
1.3实际结果描述、结论：

当拉伸窗口或最大化窗口时，窗口内所有控件能够自动调整自身的大小和位置以适应窗口，如图：

拉伸前：



拉伸后：



2.1项目名称：

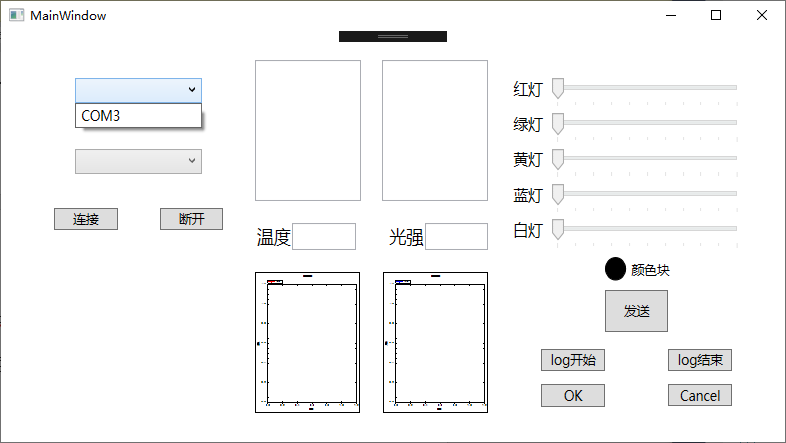
串口名及波特率的选择

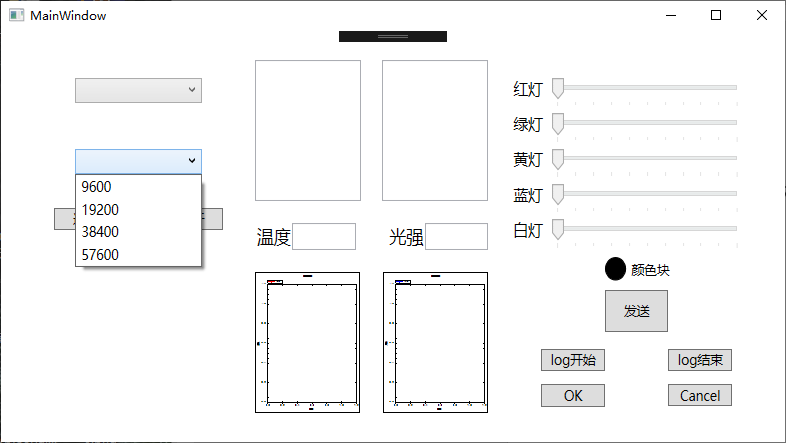
2.2操作步骤：

使用两个combobox控件用以选择串口名及波特率。其中串口名通过使用C#中SerialPort类的GetPortNames()方法取得；

2.3实际结果描述、结论：

当下拉窗口上的combobox控件时，可以选择连接的串口和波特率，如图：





3.1项目名称：

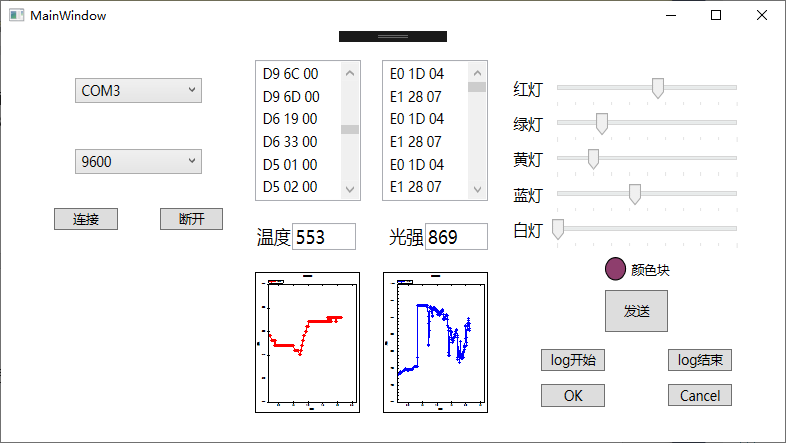
发送的数据和返回的实时信息的显示

3.2操作步骤：

使用两个ListView控件用以实现该功能，当有Arduino和PC间的数据传输时，将传输的Midi格式数据处理成字符串显示在ListView中；

3.3实际结果描述、结论：

当有数据传输时，PC接收到的数据显示在右边的ListView中，发送的数据显示在左边的ListView中，如图：



存在的问题：ListView不能自动下滑以显示最新传输的消息，需要手动下拉滚动条查看最近传输的消息；

4.1项目名称：

显示温度光强的变化曲线

4.2操作步骤：

使用ZedGraph控件以实现该功能；

4.3实际结果描述、结论：

当连接上Arduino后，左边显示温度的变化曲线，右边显示光强的变化曲线，如上图所示。

5.1项目名称：

滑块控制LED灯的亮度

5.2操作步骤：

实验中发现滑块的Value是一个10以内的double，将value处理成范围为0-255的byte类型后以Midi格式传输到Arduino；

5.3实际结果描述、结论：

连接上Arduino后，当移动滑块时，Arduino上对应的灯的亮度改变，端口接收到的信号范围是0-255；

6.1项目名称：

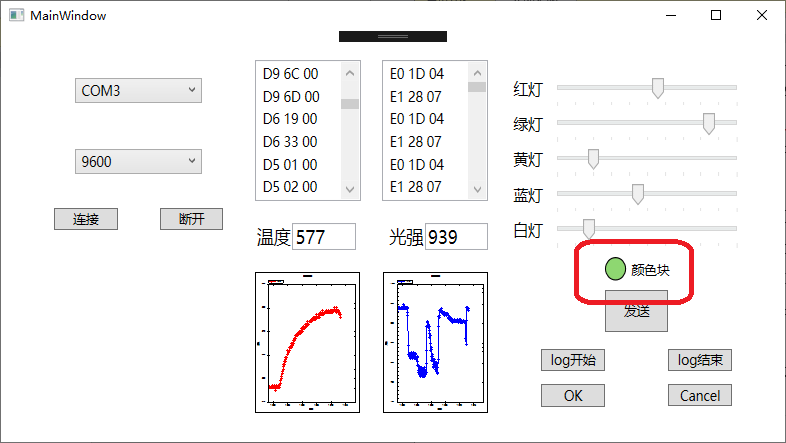
颜色块的实现

6.2操作步骤：

使用Ellipse控件完成该功能。当Slider的value发生变化时，通过各个Slider的value值计算RGB值并更新Ellipse的Fill属性以达到显示RGB混合色的效果；

6.3实际结果描述、结论：

连接上Arduino后，当移动滑块时，颜色块的RGB混合色能够灵敏的变化并显示出当前所有LED亮度值所对应的混合色，如图：



7.1项目名称：

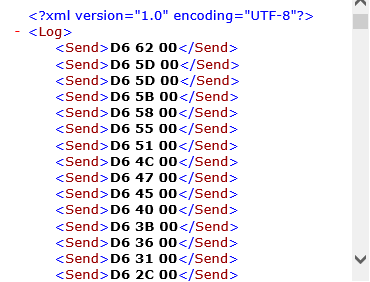
Log的存储

7.2操作步骤：

使用两个List来存储程序运行中发送和接收到的Midi消息，当用户点击了OK后，开始存储log到List中，当用户点击了Cancel时，将List清空，当点击了Log结束按钮时，将List的数据取出以Xml格式存储在硬盘上；

7.3实际结果描述、结论：

程序运行一段时间并点击结束后，PC发送和接收的log均存储在了指定的Xml文件中，如图：



## 实验结论及心得体会

通过这一次实验，我主要掌握了Arduino的Midi传输协议，知道了如何在PC上通过Midi来控制Arduino，以及通过Midi协议接收来自Arduino的信号。另外，我学会了使用xml来存储信息。除此之外，在这次实验中我尝试了使用C#中数据binding这种编程方式，发现了这种编程方式的方便之处，并最终掌握了这种binding的方式。