三分钟开发出基本的串口通信应用

1、开发环境：基于VS2010的串口通信程序，语言：ｃ#

2、方法有几个？

方法1：使用SerialPort组件，“勉强”可以。

方法2：使用ComY组件，绝对可以。

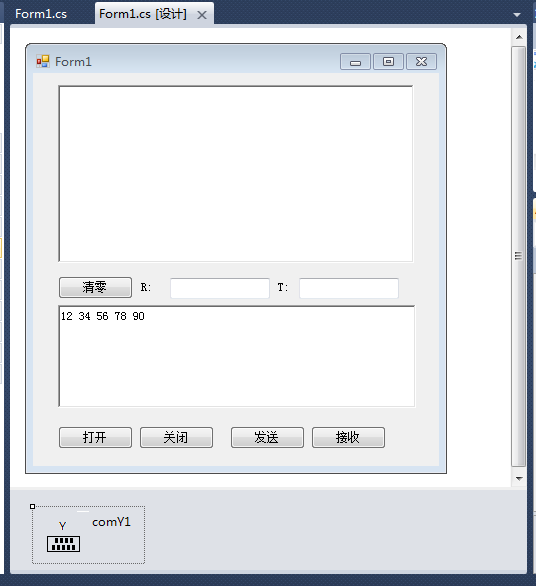
方法3：使用ComZ控件，绝对可以。

3、第一个例子（方法2）

3.1、新建项目TestComY，并拖放如下控件/组件并定义其名称如下表。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 控件类型 | 控件名称 | 用途 | 显示信息 | 备注 |
| 01 | RichTextBox | RxRichTextBox | 显示接收 |  | 接收区 |
| 02 | RichTextBox | TxRichTextBox | 输入发送数据 | 1234567890 | 发送区 |
| 03 | TextBox | RxTextBox | 显示接收数量 |  | 计数器值 |
| 04 | TextBox | TxTextBox | 显示发送数量 |  | 计数器值 |
| 05 | label | label1 | 指示接收计数区 | R | 计数器标识 |
| 06 | label | Label2 | 指示发送计数区 | T | 计数器标识 |
| 07 | Button | BtnClearCounter | 清空计数器值 | 清零 |  |
| 08 | Button | BtnComOpen | 打开串口 | 打开 |  |
| 09 | Button | BtnComClose | 关闭串口 | 关闭 |  |
| 10 | Button | BtnComTx | 发送数据区的数据 | 发送 |  |
| 11 | Button | BtnComRx | 接收并显示数据 | 接收 |  |
| 12 | comY | comY1 | 通信组件 |  | 通信服务 |

效果图如下



3.2、编写代码

双击界面中的五个按钮得到如下代码。

namespace TestComY

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void BtnComTx\_Click(object sender, EventArgs e)

{ }

private void BtnComRx\_Click(object sender, EventArgs e)

{ }

private void BtnComOpen\_Click(object sender, EventArgs e)

{ }

private void BtnComClose\_Click(object sender, EventArgs e)

{ }

private void BtnClearCounter\_Click(object sender, EventArgs e)

{ }

}

}

3.3、打开串口

private void BtnComOpen\_Click(object sender, EventArgs e)

{

comY1.SetPortPara("COM1",115200);

comY1.Open();

BtnComOpen.Enabled = false;

BtnComClose.Enabled = true;

}

3.4、关闭串口

private void BtnComClose\_Click(object sender, EventArgs e)

{

comY1.Close();

BtnComClose.Enabled = false;

BtnComOpen.Enabled = true;

}

3.5、发送数据及显示发送数据量

private void BtnComTx\_Click(object sender, EventArgs e)

{

comY1.Send(TxRichTextBox.Text);

TxTextBox.Text = comY1.TxCounterVal.ToString();

}

3.6、接收数据及显示接收数据量

private void BtnComRx\_Click(object sender, EventArgs e)

{

if (comY1.IsGotNewData() == true)

RxRichTextBox.Text += comY1.GetRxHexWithBlankString();

RxTextBox.Text = comY1.RxCounterVal.ToString();

}

3.7、清空计数器

private void BtnClearCounter\_Click(object sender, EventArgs e)

{

RxTextBox.Text = "";

TxTextBox.Text = "";

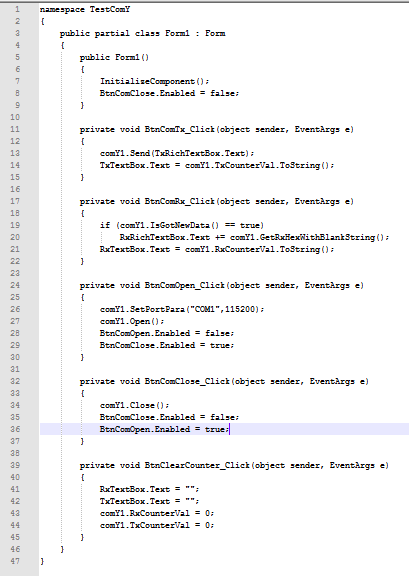
comY1.RxCounterVal = 0;

comY1.TxCounterVal = 0;

}

3.8、代码量

实现基本功能全部代码不到50行，如下图

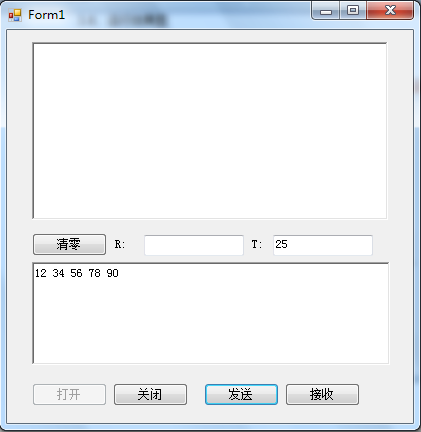


3.8、运行效果图

3.8.1、打开串口



3.8.2、发送数据



3.8.3、接收数据



3.8.4、关闭串口点发送



3.9、发送接收中文及ASCII

3.9.1、打开串口修改如下

private void BtnComOpen\_Click(object sender, EventArgs e)

{

comY1.SetPortPara("COM1",115200);

comY1.IsHexMode = false; //添加代码

comY1.Open();

BtnComOpen.Enabled = false;

BtnComClose.Enabled = true;

}

3.9.2、接收代码修改如下

private void BtnComRx\_Click(object sender, EventArgs e)

{

if (comY1.IsGotNewData() == true)

{

if (comY1.IsHexMode==true) //添加代码

RxRichTextBox.Text += comY1.GetRxHexWithBlankString();

else

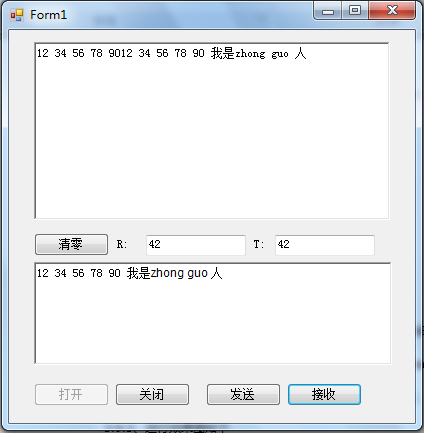
RxRichTextBox.Text += comY1.GetRxAsciiString(); //添加代码

}

RxTextBox.Text = comY1.RxCounterVal.ToString();

}

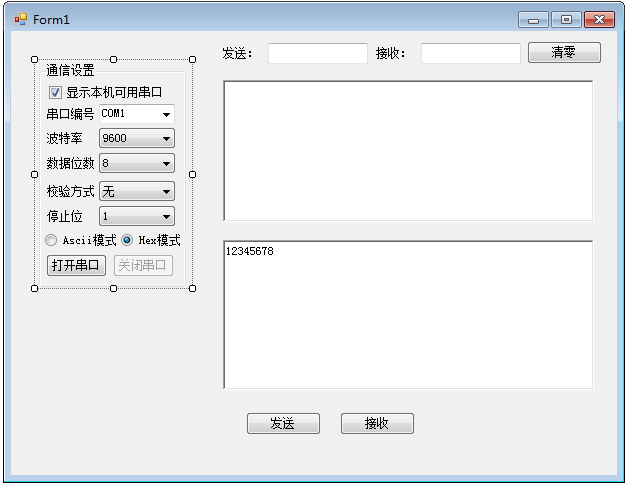
3.9.3、运行效果图如下



4、第2个例子（方法3）

4.1、新建窗体工程（testComZ）及添加控件如下表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 控件类型 | 控件名称 | 用途 | 显示信息 | 备注 |
| 01 | RichTextBox | RxBox | 显示接收 |  | 接收区 |
| 02 | RichTextBox | TxBox | 输入发送数据 | 12345678 | 发送区 |
| 03 | TextBox | RxTextBox | 显示接收数量 |  | 计数器值 |
| 04 | TextBox | TxTextBox | 显示发送数量 |  | 计数器值 |
| 05 | label | Label2 | 指示接收计数区 | 发送 | 计数器标识 |
| 06 | label | Label3 | 指示发送计数区 | 接收 | 计数器标识 |
| 07 | Button | btnClean | 清空计数器值 | 清零 |  |
| 08 | Button | btnTx | 发送数据区的数据 | 发送 |  |
| 09 | Button | BtnRx | 接收并显示数据 | 接收 |  |
| 10 | comZ | comZ1 | 通信组件 |  | 通信服务 |

效果图如下

4.2、编写代码

双击界面上的按钮得到如下代码

namespace testComZ

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void btnTx\_Click(object sender, EventArgs e)

{ }

private void BtnRx\_Click(object sender, EventArgs e)

{ }

private void btnClean\_Click(object sender, EventArgs e)

{ }

}

}

4.3、发送数据及显示发送数据量

private void btnTx\_Click(object sender, EventArgs e)

{

UInt32 txcounter;

comZ1.Send(TxBox.Text);

txcounter=comZ1.TxCounterVal;

TxTextBox.Text = txcounter.ToString();

}

4.4、接收数据及显示接收数据量

private void BtnRx\_Click(object sender, EventArgs e)

{

if (comZ1.IsGotNewData() == true)

{

if (comZ1.IsHexMode == true)

RxBox.Text += comZ1.GetRxStringAndHexWithBlank();

else

RxBox.Text += comZ1.GetRxAsciiString();

RxTextBox.Text = comZ1.RxCounterVal.ToString();

}

}

4.5、清零计数器

private void btnClean\_Click(object sender, EventArgs e)

{

comZ1.TxCounterVal = 0;

comZ1.RxCounterVal = 0;

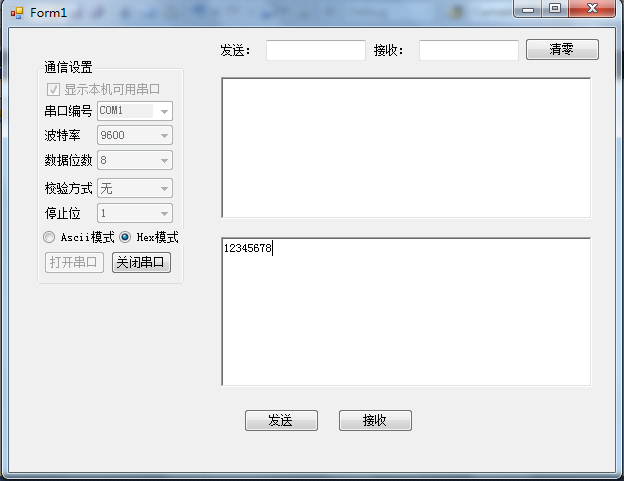
TxTextBox.Text = comZ1.TxCounterVal.ToString();

RxTextBox.Text = comZ1.RxCounterVal.ToString();

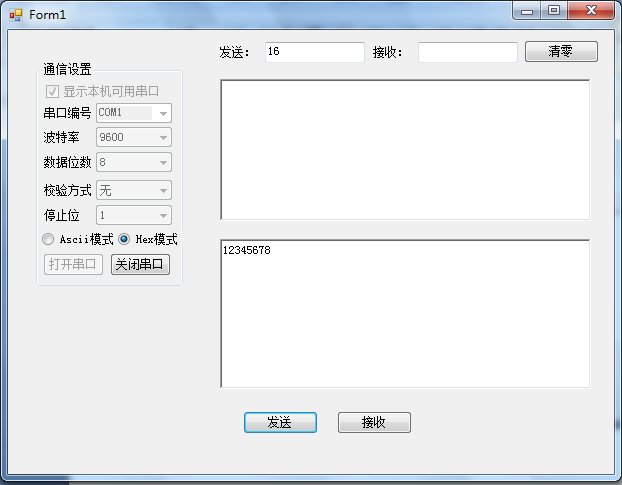
}

4.6、运行效果图

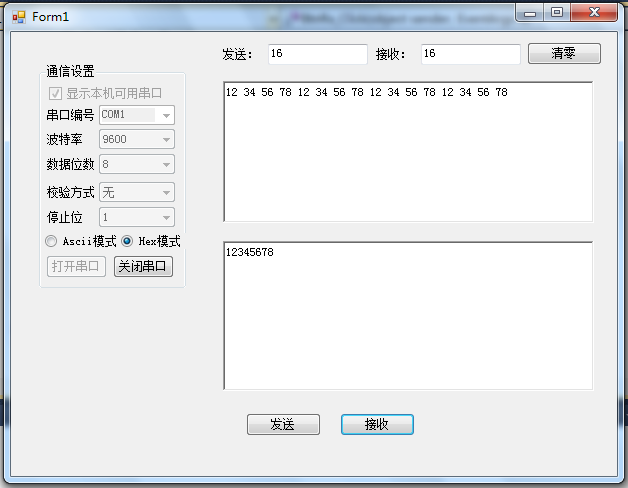
4.6.1、打开串口



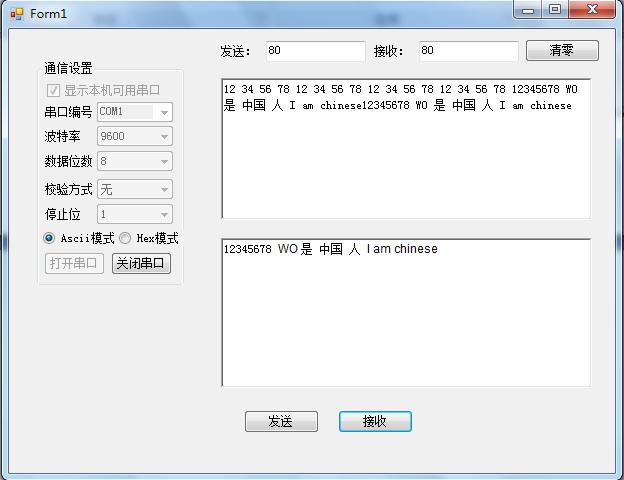
4.6.2、发送数据



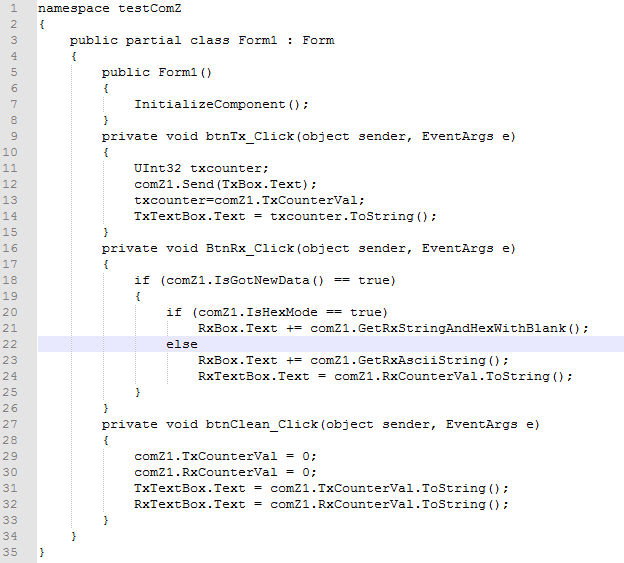
4.6.3、接收数据



4.6.4、发送中文及ASCII



4.7、代码量（全部仅仅35行）如下图



4.8、由代码量可见，整个基本功能的实现三分钟内即可完成。

5、获取测试例子，请加QQ群321903608

6、作者保留所有权限，禁止修改此文件。

7、dll文件下载

20170925