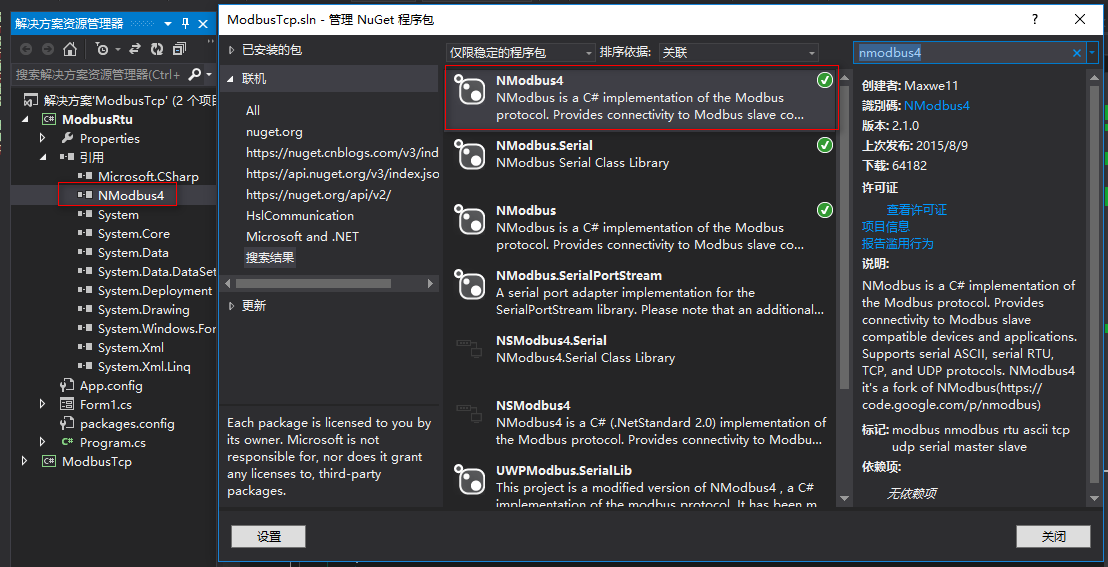
# **[C#Modbus Rtu的实现](https://www.cnblogs.com/pandefu/p/10849823.html)**

Modbus Rtu的实现与Modbus Tcp的实现类似 [C#ModBus Tcp的学习及Master的实现](https://www.cnblogs.com/pandefu/p/10824331.html)

我们还是需要借用一个开源库NModbus4,在vs中.打开NuGet管理器.安装NModbus4



具体实现,具体实现与之前的Modbus Tcp的实现类似 ,只是在实例化master时将TCPClient换为串行端口资源SerialPort,并在实例化是设置好端口所需参数(端口名,波特率,校验位,停止位,数据位)

IMG_257

[IMG_258](javascript:void(0);)

1 using System; 2 using System.Collections.Generic; 3 using System.ComponentModel; 4 using System.Data; 5 using System.Drawing; 6 using System.Linq; 7 using System.Text; 8 using System.Threading.Tasks; 9 using System.Windows.Forms; 10 using Modbus.Device; 11 using System.Net.Sockets; 12 using System.Threading; 13 using System.IO.Ports; 14 15 namespace ModbusRtu 16 { 17 public partial class Form1 : Form 18 { 19 private static IModbusMaster master; 20 private static SerialPort port; 21 //写线圈或写寄存器数组 22 private bool[] coilsBuffer; 23 private ushort[] registerBuffer; 24 //功能码 25 private string functionCode; 26 //参数(分别为站号,起始地址,长度) 27 private byte slaveAddress; 28 private ushort startAddress; 29 private ushort numberOfPoints; 30 //串口参数 31 private string portName; 32 private int baudRate; 33 private Parity parity; 34 private int dataBits; 35 private StopBits stopBits; 36 37 public Form1() 38 { 39 InitializeComponent(); 40 } 41 private void Form1\_Load(object sender, EventArgs e) 42 { 43 cmb\_portname.SelectedIndex = 0; 44 cmb\_baud.SelectedIndex = 5; 45 cmb\_parity.SelectedIndex = 2; 46 cmb\_databBits.SelectedIndex = 1; 47 cmb\_stopBits.SelectedIndex = 0; 48 } 49 private SerialPort InitSerialPortParameter() 50 { 51 if (cmb\_portname.SelectedIndex < 0 || cmb\_baud.SelectedIndex < 0 || cmb\_parity.SelectedIndex < 0 || cmb\_databBits.SelectedIndex < 0 || cmb\_stopBits.SelectedIndex < 0) 52 { 53 MessageBox.Show("请选择串口参数"); 54 return null; 55 } 56 else 57 { 58 59 portName = cmb\_portname.SelectedItem.ToString(); 60 baudRate = int.Parse(cmb\_baud.SelectedItem.ToString()); 61 switch (cmb\_parity.SelectedItem.ToString()) 62 { 63 case "奇": 64 parity = Parity.Odd; 65 break; 66 case "偶": 67 parity = Parity.Even; 68 break; 69 case "无": 70 parity = Parity.None; 71 break; 72 default: 73 break; 74 } 75 dataBits = int.Parse(cmb\_databBits.SelectedItem.ToString()); 76 switch (cmb\_stopBits.SelectedItem.ToString()) 77 { 78 case "1": 79 stopBits = StopBits.One; 80 break; 81 case "2": 82 stopBits = StopBits.Two; 83 break; 84 default: 85 break; 86 } 87 port = new SerialPort(portName, baudRate, parity, dataBits, stopBits); 88 return port; 89 } 90 } 91 /// <summary> 92 /// 读/写 93 /// </summary> 94 /// <param name="sender"></param> 95 /// <param name="e"></param> 96 private void button1\_Click(object sender, EventArgs e) 97 { 98 try 99 {100 //初始化串口参数101 InitSerialPortParameter();102 103 master = ModbusSerialMaster.CreateRtu(port);104 105 ExecuteFunction();106 }107 catch (Exception)108 {109 MessageBox.Show("初始化异常");110 }111 }112 113 private async void ExecuteFunction()114 {115 try116 {117 //每次操作是要开启串口 操作完成后需要关闭串口118 //目的是为了slave更换连接是不报错119 if (port.IsOpen == false)120 {121 port.Open();122 }123 if (functionCode != null)124 {125 switch (functionCode)126 {127 case "01 Read Coils"://读取单个线圈128 SetReadParameters();129 coilsBuffer = master.ReadCoils(slaveAddress, startAddress, numberOfPoints);130 131 for (int i = 0; i < coilsBuffer.Length; i++)132 {133 SetMsg(coilsBuffer[i] + " ");134 }135 SetMsg("\r\n");136 break;137 case "02 Read DisCrete Inputs"://读取输入线圈/离散量线圈138 SetReadParameters();139 140 coilsBuffer = master.ReadInputs(slaveAddress, startAddress, numberOfPoints);141 for (int i = 0; i < coilsBuffer.Length; i++)142 {143 SetMsg(coilsBuffer[i] + " ");144 }145 SetMsg("\r\n");146 break;147 case "03 Read Holding Registers"://读取保持寄存器148 SetReadParameters();149 registerBuffer = master.ReadHoldingRegisters(slaveAddress, startAddress, numberOfPoints);150 for (int i = 0; i < registerBuffer.Length; i++)151 {152 SetMsg(registerBuffer[i] + " ");153 }154 SetMsg("\r\n");155 break;156 case "04 Read Input Registers"://读取输入寄存器157 SetReadParameters();158 registerBuffer = master.ReadInputRegisters(slaveAddress, startAddress, numberOfPoints);159 for (int i = 0; i < registerBuffer.Length; i++)160 {161 SetMsg(registerBuffer[i] + " ");162 }163 SetMsg("\r\n");164 break;165 case "05 Write Single Coil"://写单个线圈166 SetWriteParametes();167 await master.WriteSingleCoilAsync(slaveAddress, startAddress, coilsBuffer[0]);168 break;169 case "06 Write Single Registers"://写单个输入线圈/离散量线圈170 SetWriteParametes();171 await master.WriteSingleRegisterAsync(slaveAddress, startAddress, registerBuffer[0]);172 break;173 case "0F Write Multiple Coils"://写一组线圈174 SetWriteParametes();175 await master.WriteMultipleCoilsAsync(slaveAddress, startAddress, coilsBuffer);176 break;177 case "10 Write Multiple Registers"://写一组保持寄存器178 SetWriteParametes();179 await master.WriteMultipleRegistersAsync(slaveAddress, startAddress, registerBuffer);180 break;181 default:182 break;183 }184 185 }186 else187 {188 MessageBox.Show("请选择功能码!");189 }190 port.Close();191 }192 catch (Exception ex)193 {194 195 MessageBox.Show(ex.Message);196 }197 }198 private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)199 {200 if (comboBox1.SelectedIndex >= 4)201 {202 groupBox2.Enabled = true;203 groupBox1.Enabled = false;204 }205 else206 {207 groupBox1.Enabled = true;208 groupBox2.Enabled = false;209 }210 comboBox1.Invoke(new Action(() => { functionCode = comboBox1.SelectedItem.ToString(); }));211 }212 213 /// <summary>214 /// 初始化读参数215 /// </summary>216 private void SetReadParameters()217 {218 if (txt\_startAddr1.Text == "" || txt\_slave1.Text == "" || txt\_length.Text == "")219 {220 MessageBox.Show("请填写读参数!");221 }222 else223 {224 slaveAddress = byte.Parse(txt\_slave1.Text);225 startAddress = ushort.Parse(txt\_startAddr1.Text);226 numberOfPoints = ushort.Parse(txt\_length.Text);227 }228 }229 /// <summary>230 /// 初始化写参数231 /// </summary>232 private void SetWriteParametes()233 {234 if (txt\_startAddr2.Text == "" || txt\_slave2.Text == "" || txt\_data.Text == "")235 {236 MessageBox.Show("请填写写参数!");237 }238 else239 {240 slaveAddress = byte.Parse(txt\_slave2.Text);241 startAddress = ushort.Parse(txt\_startAddr2.Text);242 //判断是否写线圈243 if (comboBox1.SelectedIndex == 4 || comboBox1.SelectedIndex == 6)244 {245 string[] strarr = txt\_data.Text.Split(' ');246 coilsBuffer = new bool[strarr.Length];247 //转化为bool数组248 for (int i = 0; i < strarr.Length; i++)249 {250 // strarr[i] == "0" ? coilsBuffer[i] = true : coilsBuffer[i] = false;251 if (strarr[i] == "0")252 {253 coilsBuffer[i] = false;254 }255 else256 {257 coilsBuffer[i] = true;258 }259 }260 }261 else262 {263 //转化ushort数组264 string[] strarr = txt\_data.Text.Split(' ');265 registerBuffer = new ushort[strarr.Length];266 for (int i = 0; i < strarr.Length; i++)267 {268 registerBuffer[i] = ushort.Parse(strarr[i]);269 }270 }271 }272 }273 274 /// <summary>275 /// 清除文本276 /// </summary>277 /// <param name="sender"></param>278 /// <param name="e"></param>279 private void button2\_Click(object sender, EventArgs e)280 {281 richTextBox1.Clear();282 }283 /// <summary>284 /// SetMessage285 /// </summary>286 /// <param name="msg"></param>287 public void SetMsg(string msg)288 {289 richTextBox1.Invoke(new Action(() => { richTextBox1.AppendText(msg); }));290 }291 292 }293 }

[IMG_259](javascript:void(0);)

接下来开始测试

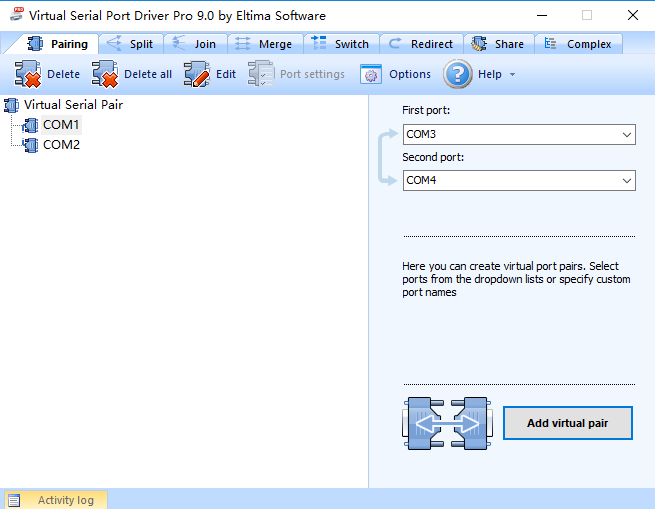
在这里 因为要用到串口,而我的笔记本没有串口,所以需要借助一个工具

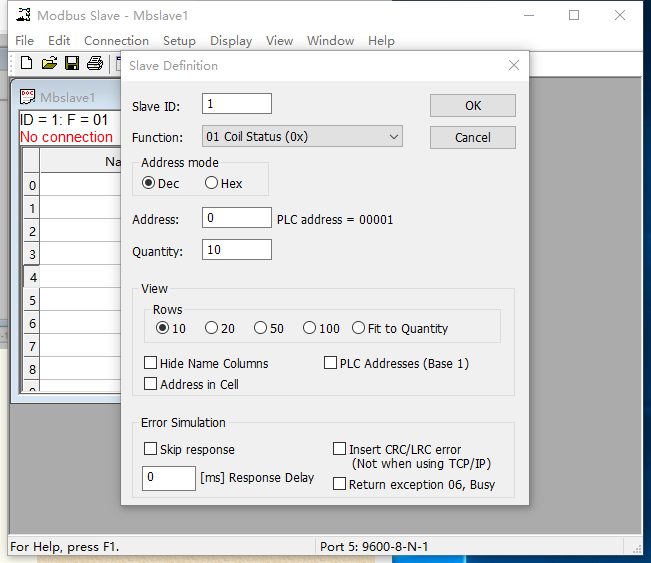
**Virtual Serial Port Dirver 虚拟串口工具**

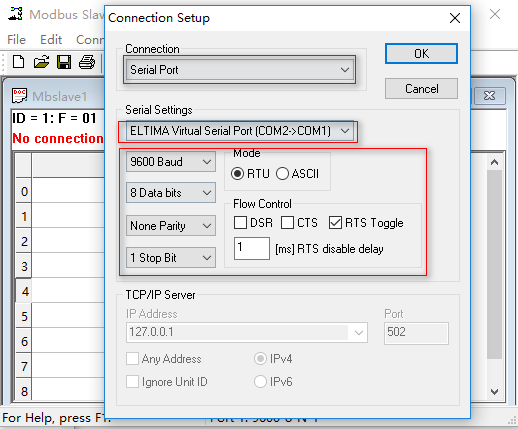
链接：[https://pan.baidu.com/s/1opGre3GS-HWFoA\_dP9qYYg](https://pan.baidu.com/s/1opGre3GS-HWFoA_dP9qYYg" \o "Virtual Serial Port Dirver 下载" \t "https://www.cnblogs.com/pandefu/p/_blank)

提取码：2afu

借用这个工具我们添加两个虚拟串口 COM1和COM2 点击Add Virtual Pair 添加



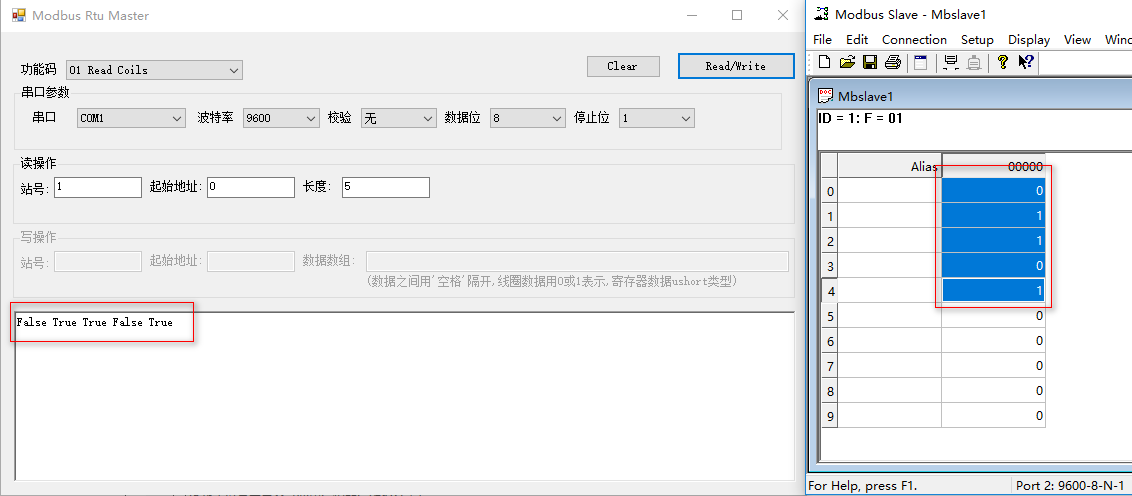
设置Modbus Slave,选择连接方式为串口,选择对应端口,模式选择RTU,建立连接



接下来运行我们自己的Modbus RTU Master

设置串口参数(波特率,数据位,奇偶校验,停止位)要与Slave的串口参数一致

我们测试 功能码 0x01 读一组线圈



测试完成,数据正常,其他的功能码经测试数据正常,有兴趣的可以自行测试

到此为止,Modbus的学习到此告一段落

以上都为我自行学习并实现,如有错误之处,望大家不吝赐教,感谢(抱拳~)

程序源代码:

链接：[https://pan.baidu.com/s/1mPAhRixLbsDb7h2ePENTRA](https://pan.baidu.com/s/1mPAhRixLbsDb7h2ePENTRA" \o "modbusDemo源代码" \t "https://www.cnblogs.com/pandefu/p/_blank)  
提取码：b5w6