基于单片机通讯的控制电机转速实验报告

1. **设计目标**

单片机通过串口1与上位机软件进行通讯，接收上位机软件发送的PID调节目标值；

单片机通过串口2与电机模型进行通讯，不断接收电机模型发送的电机当前转速值并且通过串口1将接受到的当前转速值发送给上位机软件，从而上位机软件绘制出相应的波形；

单片机基于PID调节的目标值和电机的当前转速值调用PID算法计算出PID的输出值，通过串口2发送给电机模型从而让电机模型得到新的转速；

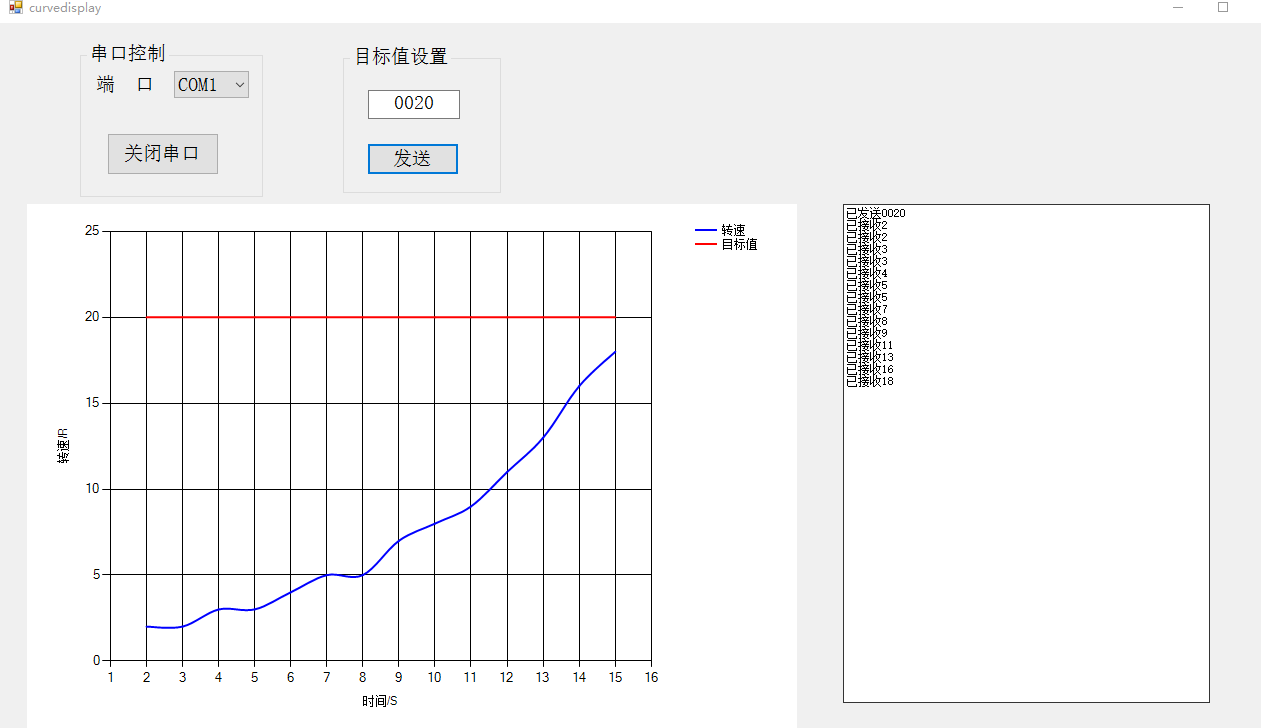
通过串口中断中置位标志位来为任务调度提供信号来实现简单的任务调度。

1. **设计内容**

本次设计分成了三部分：

2.1上位机部分

上位机采用C#来编写，作用主要是用于通过串口给单片机发送电机模型转速的PID调节目标值和接收单片机通过串口发来的电机模型的当前转速值并且将该转速值画折现图显示。如下图所示即为上位机软件的主界面。



该软件主要使用了chart控件进行曲线图的绘制，使用了serialPort进行和单片机通讯，使用了button控件进行点击控制串口的开和关，进行对目标值的发送。

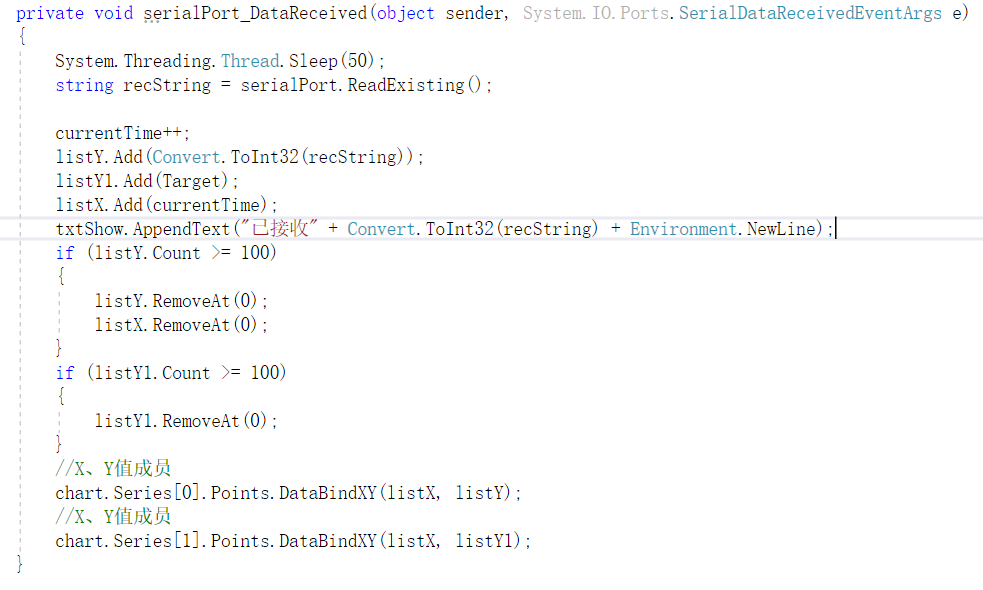
该程序功能的完成主要是三部分的代码，首先是窗口建立时候的初始化程序，主要是对

chart要绘制的曲线的初始化，代码如下图显示：

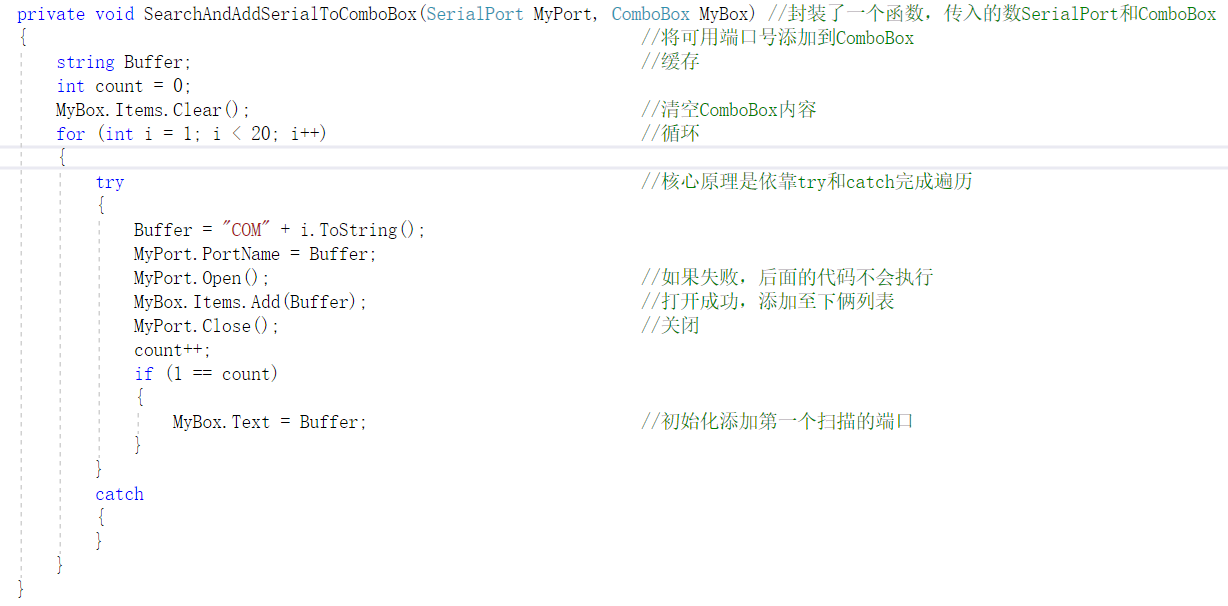
接着是上位机通过串口给单片机发送目标值的代码如下显示：



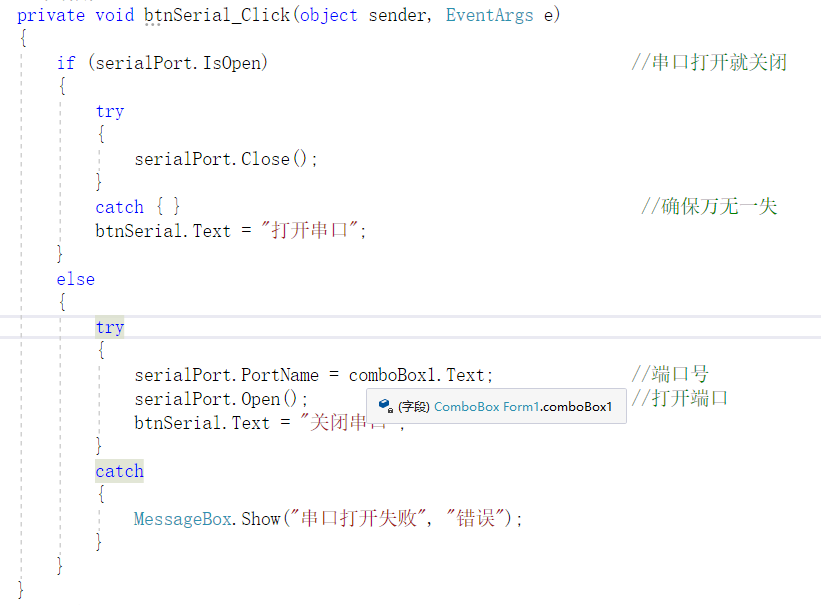
最后是串口接受到数据之后对接受到的数据处理并且绘制曲线图，还有就是为了避免曲线图过多显示点，因此设置了对曲线上的点进行判断如果超过了100个点那么就从前面开始删除，从而使得曲线图有平移的效果。代码如下图显示：



除了上面的三大部分之后还为comBox控件添加了一个点击下拉触发事件，当点击下拉的时候会触发搜索当前可以用的串口并且显示出来的方法，其代码如下显示：



此外还为打开串口设置了一个点击触发事件，如果处于关闭串口的状态那么就会打开当前扫描到并且显示的一个串口并且将文本变为关闭串口，如果处于打开串口状态那么就会关闭当前串口并且将文本改为打开串口，代码如下图显示：



2.2单片机部分

单片机部分主要是接收上位机传来的PID调节目标值，接收电机模型传来的当前转速并将当前转速发给上位机，然后不断调用PID算法得到输出值发给电机模型。由main函数开始分析，主要是三个初始化函数：

Uart1\_Init();

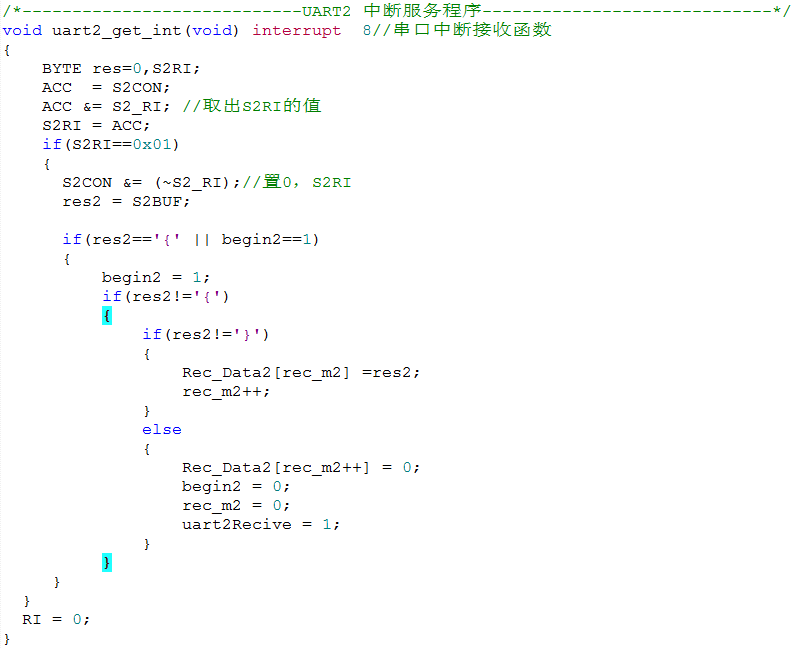
uart2\_Init();

PID\_Init();

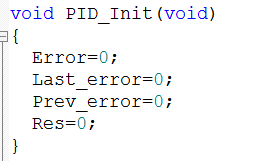
先对串口1和串口2进行初始化，并且编写相应的串口接受中断处理函数，代码入下所示



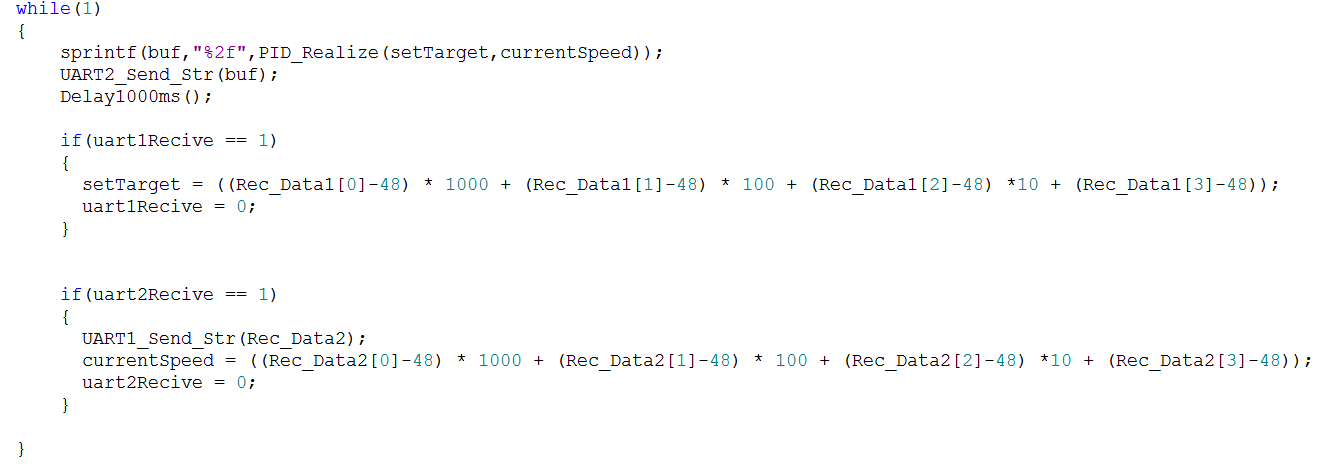


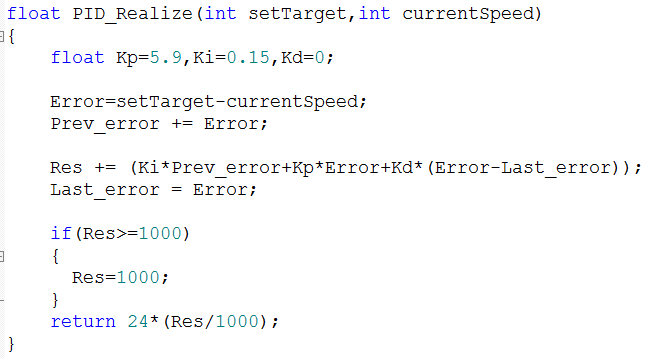


接着是对PID参数的初始化，代码入下图显示



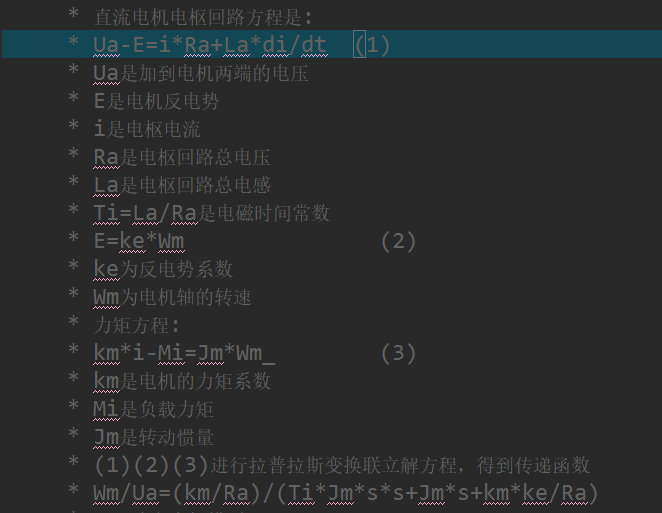
接着是主程序里面while(1)里面的代码，该代码实现了每过1000ms的时间调用一次PID算法函数进行对PID输出值的计算并且通过串口发送给电机模型，此外基于2个标志位来实现任务调度，调度的任务分别是设置PID的目标值和发送电机模型传来的当前转速值给上位机模型，这2个标志位的置位通过串口中断函数中实现置位。循环代码和PID算法代码如下显示



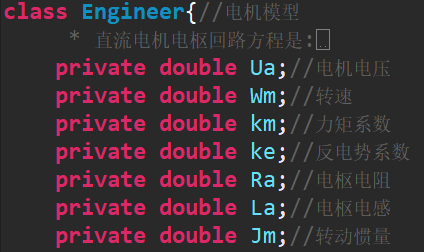


2.3电机模型

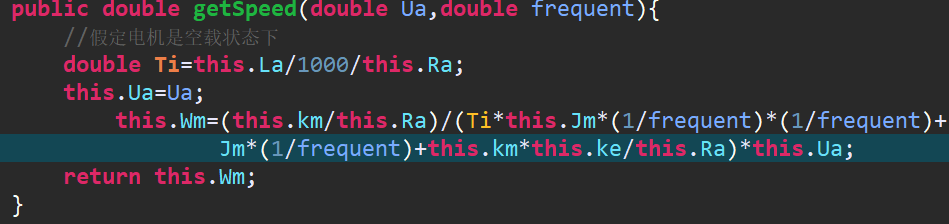
使用了Java语言编写的一个简单直流电机模型。

首先依据的电枢回路方程和力矩方程，得到电机的输入电压和转速的传递函数。

并以此建立直流电机模型电机：



以及获取电机转速，单位是 转每秒 r/s：



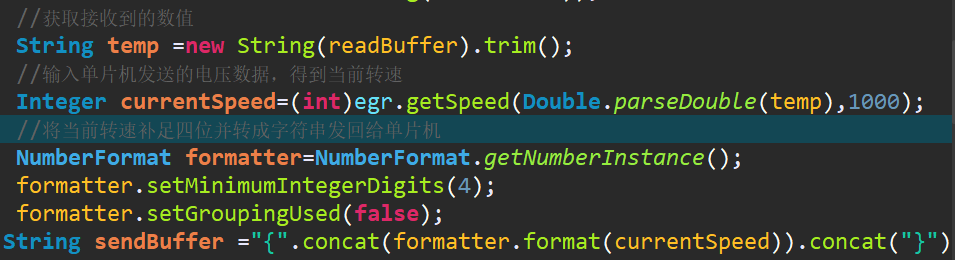
之后为电机类实例化一个电机对象，参数参考的一个小型直流电机

C:\Users\pabo\AppData\Roaming\Tencent\Users\863391618\TIM\WinTemp\RichOle\T0RYYMY5R$45@N5[0@2ZV3H.png

为了电机模型能与单片机通讯，需要用到串口，编写一个串口监听类

C:\Users\pabo\AppData\Roaming\Tencent\Users\863391618\TIM\WinTemp\RichOle\T%[(S3T_FFJ[BY@]QE99(79.png

在continueRead类里面设置串口以及读取数据并处理和发送数据。



1. **设计结果**

我们通过观看上位机软件曲线的变化来不断对PID参数进行调节，图中红线为PID调节的目标值，蓝线为电机模型当前的转速值，最后得到了较好的PID调节效果。下面的图是我们在进行PID调节时候拍摄的部分图片和拍摄视频的部分截图。

