**碳纤维原丝纺丝工段的自控系统**

**设计与实现**

**自动化1801**

**陈奕帆**

**180910117**

**目录**

[一、实验任务说明 2](#_Toc21525069)

[二、工艺要求介绍 3](#_Toc21525070)

[三、方案及设备选型的介绍 3](#_Toc21525071)

[四、相关理论概念 5](#_Toc21525072)

[五、实习实验设备 8](#_Toc21525073)

[六、DCS系统组态及调试 11](#_Toc21525074)

七、控制参数整定及效果分析 22

[八、总结 26](#_Toc21525075)

[附录 28](#_Toc21525076)

1. **实习任务说明**

碳纤维作为一种新型无机纤维材料，具有高强度、高模量、耐高温、耐腐蚀、导电、传热、热膨胀系数小等诸多优异性能，是发展航空、航天等尖端技术不可缺少的结构性材料，也是民用工业更新换代的新型材料。碳纤维的生产技术中原丝的生产技术是制造高质量碳纤维的基础，目前聚丙烯腈基纤维材料加工生产的碳纤维（PAN碳纤维），占碳纤维产量的大多数。

碳纤维生产工艺主要包括：聚合、纺前处理、纺丝、预氧化、碳化、后处理等主要环节。前三个环节称为原丝生产环节。产品为碳纤维原丝（未碳化的聚丙烯腈基纤维）。其中的纺丝工段起着将化学聚合的纺丝溶液转换为化学纤维的作用。本次专业实习选择湿法纺丝工段的三个重要流程作为典型范例，进行流程控制系统设计工程技能培训。

1．根据碳纤维原丝工艺流程（局部）控制方案的管道及仪表工艺流程图（P&ID图），设计控制系统的主要施工图（串级控制系统控制单元回路图、端子接线图、供电系统配置图、自控设备表及DCS系统I/O表）（3图2表）；

2．利用DCS系统控制系统、进行工艺设备模拟装置的传感器信号采集和处理、获取工艺状态参数。

3．结合采集的工艺参数，进行被控工艺对象的建模和控制方案的效果仿真

4．利用DCS系统控制系统、进行工艺设备模拟装置的控制系统的组态设计和控制。要求采用单回路和串级两种不同的控制方案实施控制。

5. 设计监控组态画面,完成流程控制系统的组态，实现控制系统的数据采集、查看、存储等功能。

6. 进行控制系统调试和调节器参数整定，达到控制目标

1. **工艺要求介绍**

蒸汽牵伸装置是实现PAN纤维高强化和细旦化的关键设备。牵伸时以饱和水蒸汽为牵伸介质，纤维在高压蒸汽中沿着大分子链施加张力，在高温高湿的作用下，大分子链段因热运动而产生与张力方向一致的取向，实现了均匀的可塑性牵伸，可以得到无毛丝、表面光泽的高强度、高取向度的碳纤维原丝。

在牵伸处理过程中需对牵伸装置的温度和压力加以控制，蒸汽压力过高或温度过高，会因熔融造成断丝。 为控制蒸汽箱温度，通过调节阀改变进入装置的蒸汽流量实现蒸汽箱温度调节。为避免蒸汽本身的温度波动的影响，在蒸汽进入通道用电加热器对中压蒸汽进行预热，同时为保证蒸汽牵伸箱的温度控制速度和精度，采用串级控制方案对电加热器进行操作，电加热器出口蒸汽温度为副参数，牵伸箱内温度为主参数。蒸汽箱压力则采用控制蒸汽排出量的方式进行控制。方案描述见附录P&ID图。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 工位号 | 系统、设备名称 | 控制目标 | 工艺要求 | 备注 |
| T401 | 蒸汽牵伸箱温度串级控制 | 蒸汽箱温度（TT401），  蒸汽温度（TT402） | 蒸汽箱温度（TT401）控制点：120±1℃ | 蒸汽温度波动作为影响牵伸箱温度的主要因素和主要操作量利用串级副回路加以控制。 |
| P401 | 蒸汽箱压力控制 | 蒸汽箱牵伸通道压力（P401） | 0.4Mpa |  |

表 2-1 蒸汽牵伸的工艺要求

1. **方案及设备选型的介绍**

**1.串级控制方案**

蒸汽牵伸箱温度控制系统的控制目标是牵伸箱的温度。牵伸箱温度是通过与通过蒸汽箱的蒸汽进行热交换改变的。如果采用单回路控制系统，则只有一个被控对象即牵伸箱温度。但牵伸箱温度受到蒸汽温度的波动和蒸汽箱温度两个扰动的影响。采用单回路控制系统不能很好的克服两种扰动。

为了更好的控制牵伸箱温度，选择串级控制方案。这就产生了两个被控对象：蒸汽箱温度和蒸汽温度。其中蒸汽箱温度是主对象，蒸汽温度为副对象。先对主、副对象进行建模，通过测试建模法，对对象加上阶跃信号得到被控对象的时间常数T和滞后时间常数t。以此为依据进行建模。

主控制器和副控制器均选择PID控制器。主控制器的输出作为副控制器的输入。先对副对象加PID控制器进行调节，使温度响应曲线具有10：1的衰减率并且快速稳定。然后保持副控制器的参数不变对主控制器进行整定，通过调节主控制器的PID参数使温度相应曲线能够以衰减震荡的形式逐步稳定到设定的温度。

**2.设备选型介绍**

**测量部分：**

1. 一体化Pt100热电阻温度传感器

设备型号：WZB-841-GS；传感器信号：变送信号1~5V，供电DC24V

1. 8路12位模拟量输入模块

设备型号：CPSN-AI-1208LI；信号类型：8通道电压信号输入，DC 1~5V 信号输入

1. 安全保护器

设备型号：KFD2-VR-Ex1.18；信号类型：AI连接输入DC 0~12V输出DC 0~12V

**控制部分：**

1. PAC

设备型号：CONTEC CPS-PC341MB-ADSC1-9201

在工控机上进行编程，通过PID等算法对温度进行控制。

**执行部分：**

1. 2路12位模拟量输出模块

设备型号：CPSN-AO-1602LC；信号类型：2通道，DC4~20mA输出

1. 安全保护器

设备型号：KFD2-CD-Ex1.32；信号类型：AO连接输入DC 4~20mA输出DC 4~20mA

1. 固态继电器

设备型号：JGX-3 032 3840Z；信号类型：控制电压3-32VDC 控制电流<32mA

1. 电加热器

设备型号：DN40；控制电压：220V/380V；加热功率：3KW~12KW

**3.图纸设计**

根据工程设计需要，本次实习共完成了以下几部分的工艺和施工图纸设计。包括温度串级仪表回路图、AIAO端子接线图、控制柜供电布置图以及DCS系统表和自控设备表。共三张图两张表格。其中：

温度串级仪表回路图：按P&ID图形式绘制仪表符号和标注，布置在不同区域内。在仪表符号周围绘制使用的接线端子，绘制连线使用的端子排端口，进行连线。该图根据蒸汽牵伸箱温度压力控制方案的PI&D图绘制。绘制了从现场到端子柜、机架到操作台的接线图。

AIAO端子接线图：端子接线图是在一张图纸上将DCS系统某个控制柜中的所有同类功能信号传送模块和端子排的连接关系全部表达出来，通过“相对呼应接线法”表示端子和DCS设备之间或端子之间的连线关系。该图绘制了仪表回路图的现场到端子柜的具体接线和设备与AIAO端口的接线。

控制柜供电布置图：通过绘图说明控制柜供电的方案设计，主要用以进行控制柜交直流供电的施工布置和接线。图中标明交流电源、直流电源的连接，并配以相应的低压开关（断路器、保护器、熔断器等）、对于供电的电源接线端子布置。列表说明使用的低压设备的型号、名称等信息。

DCS系统I/O表：根据设计要求和控制流程、采集流程及工艺流程的需要，仔细地列出所有的控制及采集信号。

自控设备表：根据实际系统的设备选择，列出所用设备仪表的型号、规格、数量等信息。

1. **相关理论概念**
2. **PID比例积分微分控制**

（1）PID控制的基本概念:一般的PID控制系统原理如所示，这是一个典型 PID控制系统原理框图,单位负反馈控制系统，系统由PID控制器和被控对象组成。



图4-1 PID控制系统原理图

PID控制器是实现对输入的偏差信号进行比例、积分、微分运算的器件。

输入：控制偏差e( t ) = r( t ) - y( t )

输出：偏差的比例(P)、积分(I)和微分(D)的线性组合

（2）基本的PID控制规律可描述为：



图4-2 PID控制规律

式中 KP —— 比例系数

KI —— 积分时间常数

KD —— 微分时间常数

（3）PID各环节作用：

比例环节：即时成比例地反应控制系统的偏差信号e(t)，偏差一旦产生，控制器立即产生控制作用以减小误差。当偏差e=0时，控制作用也为0。因此，比例控制是基于偏差进行调节的，即有差调节。

积分环节：能对误差进行记忆，主要用于消除静差，提高系统的无差度，积分作用的强弱取决于积分时间常数Ti，Ti越大，积分作用越弱，反之则越强。

微分环节：能反映偏差信号的变化趋势(变化速率)，并能在偏差信号值变得太大之前，在系统中引入一个有效的早期修正信号，从而加快系统的动作速度，减小调节时间。

1. **串级控制系统**

串级控制系统由两个控制器组成，一个控制器的输出作为下一个控制器的设定值，两个控制器各有自己独立的测量输入，只有主控制器有独立的设定值，副控制器的输出传送给被控变量。

传统的反馈控制是在被控变量和设定值之间产生偏差之后才起作用。如果干扰不可测或者无法获得扰动通道的数学模型，就不能采用反馈控制策略。串级控制通过选择第二个测量点构成第二个反馈回路来克服干扰。第二个测量点应该比被控变量更快感知到干扰的影响，这样才能在干扰对被控变量产生很大影响之前通过第二个反馈回路迅速克服干扰的影响。

与单回路控制系统相比，串级控制系统能够迅速克服进入副回路的二次干扰，减小了其对主变量的影响，改善了控制效果。

在一定程度上提高了对一次干扰的抗干扰能力，这是因为由于副回路的存在，大大减小了副回路的时间常数，对于主回路而言，其控制通道缩短了，因而克服一次干扰的能力仍高于同等条件下的单回路控制系统。

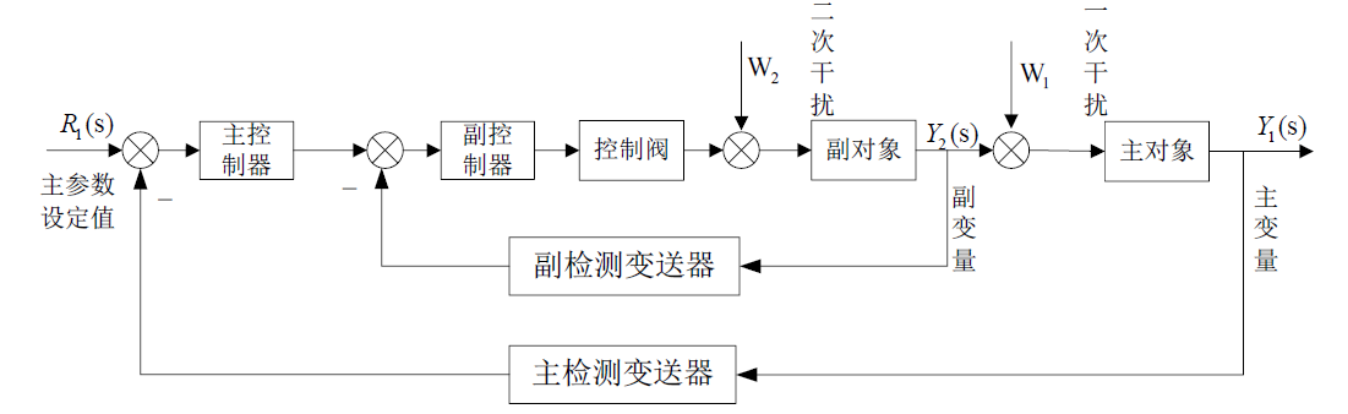


图 4-3 串级控制系统

1. **数据滤波**

在数据测量上，数据滤波是指：在对实物扫描数字化的过程中，会不可避免的引入错误点、冗余点以及扫描环境所带来的测量噪声等，这些点对后期的实物模型重构会带来很大影响，为了更好的提取实物的特征数据，必须进行数据滤波，将这些错误点等进行去除。通常的数据滤波主要是借助数字信号处理和图像处理中的滤波技术来进行，但也有一些研究者采用了一些自己的创新方法来进行数据滤波，达到了很好的滤波效果。

由于在温度的测量过程中经常会有各种扰动使得温度采样数据出现较大的波动。反映在图像上就是曲线的毛刺较多。需要进行数据滤波使曲线变得平滑，也利于后面整定PID参数和最终结果的稳定。常见的数据滤波方法有：限幅滤波、中值滤波、算术平均滤波等方法。

限幅滤波：首先根据经验判断，确定两次采样值允许的最大偏差值（设为A），每次检测到新值时判断：如果本次值与上次值之差≤A,则本次值有效；如果本次值与上次值之差＞A，则本次值无效，放弃本次值，用上次值代替本次值。本次实验对热敏电阻计算得到的温度变化进行限幅，上一时刻的温度为T0 ，则当前时刻的温度T1 若超出T0±A的温度范围，则将当前时刻温度T1限制为T0±A，这样既能对温度进行限幅，也能保证温度升高或者降低的变化趋势。

算数平均滤波：算数平均滤波就是对采样值进行N次采样然后取平均值来减小过大或者过小的采样值的不利影响。由于实验设备易受外界影响产生温度变化导致分压电路所得电压U0 跳变。在程序中使用中值滤波，对计算所得的温度值进行100次采样并且累加起来得到累加值sum，然后将sum/100取平均，将得到的平均值作为当前温度。

1. **实习实验设备**

由于在实验室环境中，不具备工业现场的真实设备和环境，在专业实习中，采用与工业设备控制对象机理相近的实验装置模拟实际工业环境的温度变化。

1. 温控箱

模拟串级系统副回路的温度测量及操作装置，传感器输出信号0~10VDC，模拟电加热器出口蒸汽温度传感器信号，作为串级系统的副参数，对应温度变化范围（0~200℃）。操作装置输入电压0~10VDC，控制温度变化。本次设计实验中该装置作为温度控制对象副回路设备，传感器信号类型DC 0~10V。控制信号类型DC 0~10V。

1. 热敏电阻传感器及信号变送电路

输出电压2.5~4VDC，对应于热敏电阻测温电路输出信号，实验中将实测温度扩大3倍模拟蒸汽牵伸箱的内部温度变化，作为串级系统的主参数

主回路设备的温度测量由热敏电阻实现，实验采用的热敏电阻为NPC型热敏电阻，阻值随测量温度的上升而下降。由于热敏电阻阻值受温度影响变化明显，电阻信号转换电路采用了简单的直流分压电路。电路由热敏电阻和阻值固定电阻串联组成，以5VDC电压供电，电路输出Uo为分压电路中固定电阻的分压值。

热敏电阻参数为：R25=10000Ω， B=3950K。

固定电阻阻值：R=10000Ω。

转换电路原理图见图5-1。在25℃时，热敏电阻温度值为10KΩ，电路输出为2.5V。当热敏电阻测量的温度大于25℃时，由于热敏电阻的阻值下降，固定电阻的分压Uo上升。可以通过电压大小，确定热敏电阻的阻值，进而计算出实际温度。实验中将实测温度扩大3倍模拟蒸汽牵伸箱的箱内温度。

实验中以温控箱的加热器对粘附有传感器的铁块进行加热，通过所设计的控制系统对铁块表面温度进行控制。

热敏电阻的阻值温度特性，可用以下的非线性函数表示：

公式中的热敏指数B值是NTC热敏电阻的属性之一，本实验采用的热敏电阻的系数B=3950。

公式中的温度T0和T1采用热力学温标表示与摄氏温度的转化关系为：

*T*0（热力学温度）=273.15+*t0*（摄氏温度）

*R*T0为特定常温下的热敏电阻阻值，本实验采用的热敏电阻的RT0为25℃时10。

计算可得到摄氏温度的计算公式为：

X=1/(273.15+25); Y=; B=3950;

摄氏温度 =.

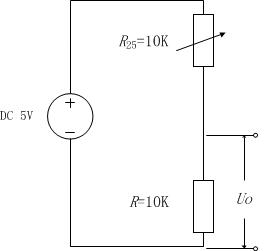


图 5-1 热敏电阻分压电路

1. PAC控制器

PAC控制器内置CODESYS软PLC引擎，主要负责采集各种IO信息，对实验过程进行控制，通过OPC-UA协议实现和服务器的交互，接收服务器的信息，并把实验数据发送到服务器。

远程I/O采集单元(简称NANO)，内置RS-232C串口和网口，四片可插拔式IO子板，可根据用户的需要自由配置。支持MODBUS/TCP协议和上位控制器交换数据。本平台配置8路-10V～10V电压输入、2路-10V～10V电压输出、8点DI输入和8点DO输出。

PAC控制器的A网口的IP地址出厂设置为172.168.1.XX，指定与服务器通信，与操作电脑通信。（XX根据实验台的编号决定），服务器、操作电脑和PAC控制器的A网口需配置为同一LAN网段。

PAC控制器的B网口的IP地址设置为192.168.1.101，指定与NANO进行通信。每个实验台的NANO模块IP地址均配置为192.168.1.103

1. **DCS系统组态及调试**
2. **信号测量**
3. **硬件配置结果**

先配置好PAC控制器的网络端口，再添加主从服务器。修改主从服务器的IP地址。建立相应的I/O通道并配置好。

实验设备中需要配置硬件模拟输入模块CPSN-AI-1208LI和模拟输出模块CPSN-AO-1602LC的端口，共用到了模拟输入的两个的端口和模拟输出的一个端口。模拟输入的两个channel用来接收热敏电阻分压电路的电压，用于在codesys软件中编程计算出热敏电阻测得的主对象温度。模拟输出的一个channel用于输出在算法处理得出的加热电压或者手动操作设定值对应的输出电压，用来控制加热箱的加热。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 变量 | 通道 | 地址 | 类型 | 描述 |
| AI00 | AI Channal 2[0] | %IW1 | WORD | 读取对象温度值，并转化为模拟量输入 |
| AI01 | AI Channal 2[1] | %IW2 | WORD | 读取对象温度值，并转化为模拟量输入 |
| AD01 | AD Channal 3[0] | %QW1 | WORD | 输出温度控制信号 |

表 6-1 硬件配置结果

1. **测量数据处理程序**

热敏电阻分压的电压以及测得的加热箱温度都是模拟量，通过将这些模拟量转化成数字量之后才能传入系统中。在编程的过程中，我们还要编写相应的代码将这些数字量转化成模拟量进行运算处理，输出时还要将计算得到的模拟量转化成数字量进行输出。

数据处理部分主要包括对采集到的副对象铁片的温度进行数模转换部分，将热敏电阻分压电采集的电压利用给出的热敏电阻的阻值温度特性来计算出模拟的主对象铁块的温度部分。以及对PID算法的输出或手动操作的设定值进行模数转换作为输出的加热电压部分。对采集到的电压数据进行数据滤波提高精度部分。

变量定义部分：  
PROGRAM PLC\_PRG

VAR

TempData1,TempData2:WORD;

//温控箱温度对应电压，电阻对应电压

CtrlOut:WORD; //输出电压

MT,Uo,NT,NTemp,OP1,OP2,op,x,y:REAL;

//副对象温度、电压输入Uo、主对象温度、主对象温度中间变量、主PID模块输出、副PID模块输出、操作输出、计算热敏电阻温度的中间变量x、y

System\_Start:BOOL; //加热开关

NT0:REAL:=0; //上一时刻主对象温度

count:BYTE; //计数

a:REAL; //中间变量；存储分压电路电压

P1,P2:REAL; //P1,P2为主、副控制器的比例系数

I1,I2,Z:REAL; // I1,I2为主、副控制器的积分系数

D1,D2:REAL; // D1,D2为主、副控制器的微分系数

SV,MOP:REAL; //SV设定值，MOP手动操作输出

amSwitch,reset,war1,war2,war3,war4:BOOL;

END\_VAR

程序部分：  
//计数部分，用于平均值滤波

count:=count+1;

IF count>100 THEN

count:=1;

END\_IF

//温度计算部分

MT:=200\*(TempData1-2048)/2048.0;

Uo:=10\*(Tempdata2-2048)/2048.0;

x:=1/298.15;

y:=(LN((5-Uo)/Uo))/3950;

NTemp:=3\*(1/(X+Y)-273.15);

//MT为模拟的工业温度，先从数字量转化成模拟量再x200变为实际温度；再由电压计算得到热敏电阻测得的温度，NTemp为滤波前计算得到的工业摄氏温度。

//滤波算法部分

IF count<=100 THEN

a:=NTemp+a;

IF count =100 THEN

NT:=(a/100);

IF NT0<1 THEN

nt0:=NT;

ELSE

IF NT>NT0+0.5 THEN

NT:=NT0+0.5;

END\_IF

IF NT<NT0-0.5 THEN

NT:=NT0-0.5;

END\_IF

nt0:=NT;

END\_IF

a:=0;

END\_IF

END\_IF

//滤波算法使用了算数平均值滤波和限幅滤波，通过计数使得NTemp主温度计数100次平均后再赋给NT，减少采样电压变化过大引起图像的跳变。并且对NT进行限幅，NT变换相比上一次结果波动大于0.5时，限制NT变化幅度为0.5

1. **测量效果**

使用数据滤波前：

由图像可以看出，在手动模式下加热时，虽然整体上对象的温度是上升的趋势，但是图像的波动剧烈，不利于后面进行PID参数整定，需进行

数据滤波。



图 6-1 数据滤波前

使用数据滤波后：

使用均值加限幅滤波后，可以看到曲线变得比之前平滑很多，图像不会产生剧烈的温度跳变。



图 6-2 数据滤波后

1. **算法组态**

程序由一个主程序PLC\_PRG和两个PID控制块POU和POU\_1组成。

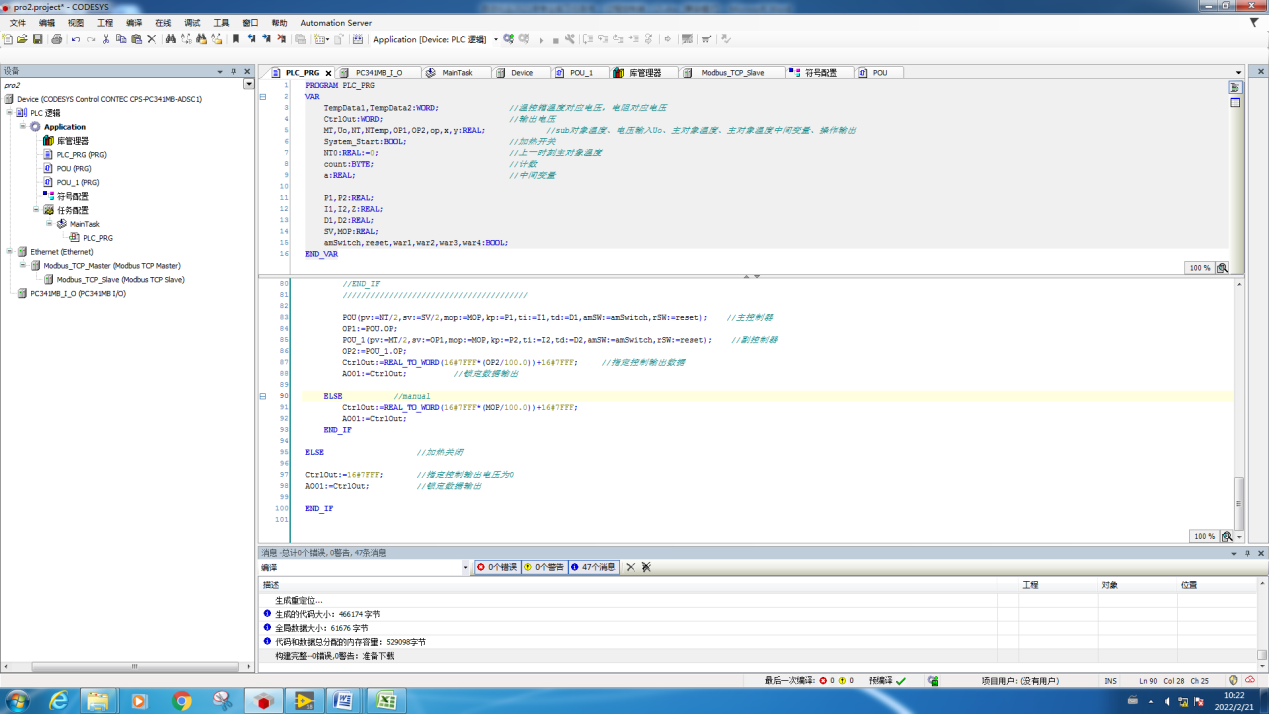


图 6-3 算法组态界面展示图

主程序进行变量定义，输入输出数据的处理，算法的调用。

两个PID块用于接收主副控制器的PID参数并调用PID算法进行计算。

|  |  |  |  |
| --- | --- | --- | --- |
| 变量 | 含义 | 类型 | 作用 |
| TempData1 | 温控箱温度对应电压 | WORD | 保存AI通道1端口的数据 |
| TempData2 | 电阻对应电压 | WORD | 保存AI通道2端口的数据 |
| CtrlOut | 输出电压 | WORD | 获取输出控制电压 |
| MT | 副对象温度 | REAL | 显示副对象实际温度 |
| Uo | 电压输出Uo | REAL | 获取固定电阻的分压 |
| NT | 主对象温度 | REAL | 显示滤波后主对象温度 |
| NTemp | 主对象温度中间变量 | REAL | 获取滤波前每一个主对象温度 |
| OP1 | 主IPD模块输出 | REAL | 获取主IPD模块输出 |
| OP2 | 副IPD模块输出 | REAL | 获取副IPD模块输出 |
| op | 操作输出 | REAL | 保证手自动切换顺滑 |
| x | 计算热敏电阻温度的中间变量 | REAL | 热敏电阻温度计算的中间值 |
| y | 计算热敏电阻温度的中间变量 | REAL | 热敏电阻温度计算的中间值 |
| System\_Start | 加热启动开关 | BOOL | 加热启动变量 |
| NT0 | 上一时刻主对象温度 | REAL | 限幅滤波所需变量 |
| count | 计数变量 | BYTE | 均值滤波所需变量 |
| a | 滤波中间变量 | REAL | 100个主对象温度数据的均值 |
| P1 | 主PID模块比例系数 | REAL | PID模块中决定比例作用强弱的系数 |
| P2 | 副PID模块比例系数 | REAL | PID模块中决定比例作用强弱的系数 |
| I1 | 主PID模块积分系数 | REAL | PID模块中决定积分作用强弱的系数 |
| I2 | 副PID模块积分系数 | REAL | PID模块中决定积分作用强弱的系数 |
| D1 | 主PID模块微分系数 | REAL | PID模块中决定微分作用强弱的系数 |
| D2 | 副PID模块微分系数 | REAL | PID模块中决定微分作用强弱的系数 |
| SV | 设定值 | REAL | 期望的最终控制对象数值 |
| MOP | 手动操作输出变量 | REAL | 手动输出功率数值 |
| amSwitch | 手自动切换变量 | BOOL | 自动模式与手动模式的切换变脸 |
| reset | PID绑定变量 | BOOL | PID重置变量 |
| war1 | PID绑定变量 | BOOL | 判断主PID模块输出值是否溢出给定范围 |
| war2 | PID绑定变量 | BOOL | 判断主PID积分模块是否运算溢出 |
| war3 | PID绑定变量 | BOOL | 判断副PID模块输出值是否溢出给定范围 |
| war4 | PID绑定变量 | BOOL | 判断副PID积分模块是否运算溢出 |

表 6-2 程序变量注释表

变量定义程序：

PROGRAM PLC\_PRG

VAR

TempData1,TempData2:WORD; //温控箱温度对应电压，电阻对应电压

CtrlOut:WORD; //输出电压

MT,Uo,NT,NTemp,OP1,OP2,op,x,y:REAL; //副对象温度、电压输入Uo、主对象温度、主对象温度中间变量、主PID模块输出、副PID模块输出、操作输出、计算热敏电阻温度的中间变量x、y

System\_Start:BOOL; //加热开关

NT0:REAL:=0; //上一时刻主对象温度

count:BYTE; //计数

a:REAL; //中间变量；存储分压电路电压

P1,P2:REAL; //P1,P2为主、副控制器的比例系数

I1,I2,Z:REAL; // I1,I2为主、副控制器的积分系数

D1,D2:REAL; // D1,D2为主、副控制器的微分系数

SV,MOP:REAL; //SV设定值，MOP手动操作输出

amSwitch,reset,war1,war2,war3,war4:BOOL;

//自动手动开关，PID复位，主控制器PID的输出限幅、溢出信号；副控制器PID的输出限幅、溢出信号

END\_VAR

主程序：

//算数平均值滤波累加计数

count:=count+1;

IF count>100 THEN

count:=1;

END\_IF

//数据传递

TempData1:=AI00; //将AI通道第1端口数据传送给TempData1变量，副对象

TempData2:=AI01; //将AI通道第2端口数据传送给TempData2变量，主对象

//主、副对象温度计算数据处理

MT:=200\*(TempData1-2048)/2048.0;

Uo:=10\*(Tempdata2-2048)/2048.0;

x:=1/298.15;

y:=(LN((5-Uo)/Uo))/3950;

NTemp:=3\*(1/(X+Y)-273.15);

//获取PID算法子程序输出

op2:=pou\_1.OP;

war1:=pou.W1;

war2:=pou.W2;

war3:=pou\_1.W1;

war4:=pou\_1.W2;

//PID运算溢出的处理

IF WAR2=TRUE THEN

RESET:=TRUE;

END\_IF

IF WAR4=TRUE THEN

RESET:=TRUE;

END\_IF

//无扰动手自动切换处理

IF amswitch=TRUE THEN

op:=mop;

op2:=op;

ELSE

op:=op2;

mop:=op;

END\_IF

//限幅和算术平均值滤波

IF count<=100 THEN

a:=NTemp+a;

IF count =100 THEN

NT:=(a/100);

IF NT0<1 THEN

nt0:=NT;

ELSE

IF NT>NT0+0.5 THEN

NT:=NT0+0.5;

END\_IF

IF NT<NT0-0.5 THEN

NT:=NT0-0.5;

END\_IF

nt0:=NT;

END\_IF

a:=0;

END\_IF

END\_IF

IF System\_Start=TRUE THEN //加热开启

IF NOT amswitch THEN //auto

POU(pv:=NT/2,sv:=SV/2,mop:=MOP,kp:=P1,ti:=I1,td:=D1,amSW:=amSwitch,rSW:=reset); //主控制器

OP1:=POU.OP;

POU\_1(pv:=MT/2,sv:=OP1,mop:=MOP,kp:=P2,ti:=I2,td:=D2,amSW:=amSwitch,rSW:=reset); //副控制器

OP2:=POU\_1.OP;

CtrlOut:=REAL\_TO\_WORD(16#7FFF\*(OP2/100.0))+16#7FFF; //指定控制输出数据

AO01:=CtrlOut; //锁定数据输出

ELSE //manual

CtrlOut:=REAL\_TO\_WORD(16#7FFF\*(MOP/100.0))+16#7FFF;

AO01:=CtrlOut;

END\_IF

ELSE //加热关闭

CtrlOut:=16#7FFF; //指定控制输出电压为0

AO01:=CtrlOut; //锁定数据输出

END\_IF

1. **HMI功能设计**

人机交互界面是通过Labview进行设计的由前面板和程序框图组成的。

前面板使用的控件有数据输入控件、数据显示控件、布尔开关、图表显示控件、文件存储控件。

前面板共8个数据输入控件，其中3个用于修改系统主控制器的PID参数P1、I1、D1；3个用于修改副控制器的PID参数P2、I2、D2。1个用于设置自动模式下温度的设定值SV，一个用于设置手动模式下的手动输出MOP。

两个显示控件分别用来显示模拟系统主对象铁块即热敏电阻测得的温度NT，副对象即加热箱铁片的温度MT。

一个布尔开关空间用于加热系统自动手动模式的切换，一个布尔开关start用于系统的开启和关闭。一个停止按键用于停止while循环。

文件存放输入框用来设置波形图数据的存放路径，需要在程序执行前设置好。

一个波形图表可以帮助我们更加直观的观察温度在长时间的变换情况。

程序框图中列出了这些变量的连接情况。主副控制器的PID参数、手动操作输出MOP、设定值SV、自动手动模式切换开关、系统开关为只写形式。主对象温度NT、副对象温度MT为只读形式。在OPC设定时要将这些变量的只读/只写设置好。为了将前面板波形图表的数据保存到excel中，需要引入获取当前时间等控件，将需要的数据如主对象温度、副对象温度、设定值等链接在一起构成一个数组输出到excel表格中，便于后续的处理。

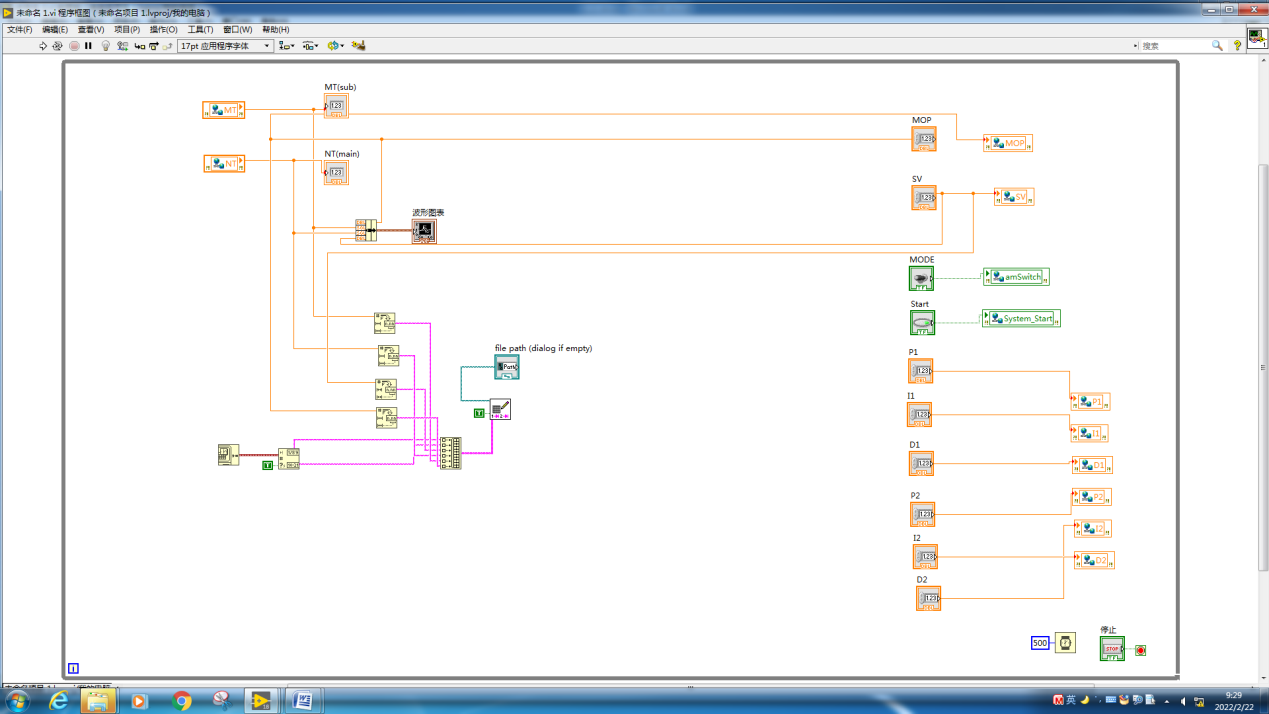


图 6-4 程序框图



图 6-5 前面板图

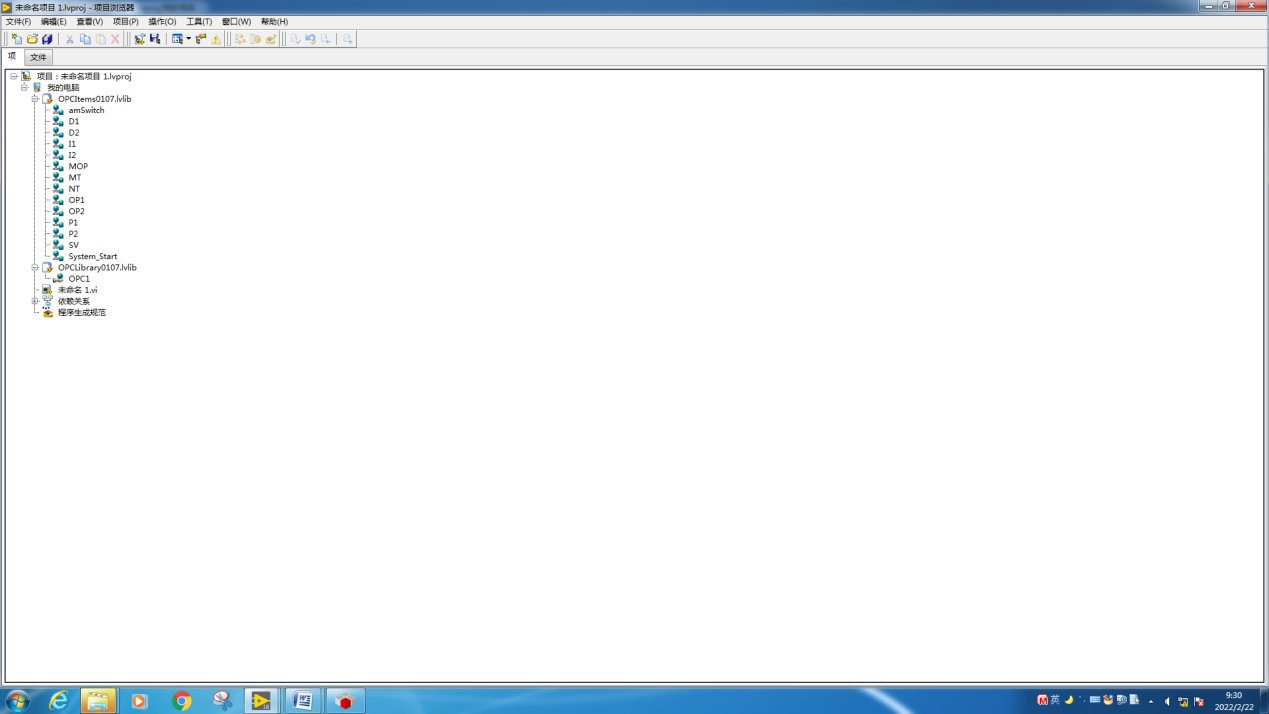


图 6-6 OPC通信配置图

1. **控制参数整定及效果分析**
2. **建模**

将对象看作自衡模型，给已经在当前温度下稳定的对象加阶跃信号直到达到新的平衡状态，在去掉阶跃信号，等对象降温至另一个稳定状态。

通过分别对主、副对象加阶跃信号采用测试建模法得到图像如下图。

计算增益K：

如阶跃输入u(t)的变化幅值为Δu(t)，则增益K仍根据输入/输出稳态值的变化来计算，即

其中，y(0)和y(∞)分别为输出y(t)的起始值和稳态值。

计算参数T和τ：

首先需要把输出y(t)转换成它的无量纲形式 y\*(t)，即



系统化为无量纲形式后，所对应的传递函数可表示为

一般选取在t1和t2时刻的输出信号分别为 y\*(t1)=0.39，y\*(t2)=0.63，此时由式可得T=2(t2-t1)，τ=2t1-t2。

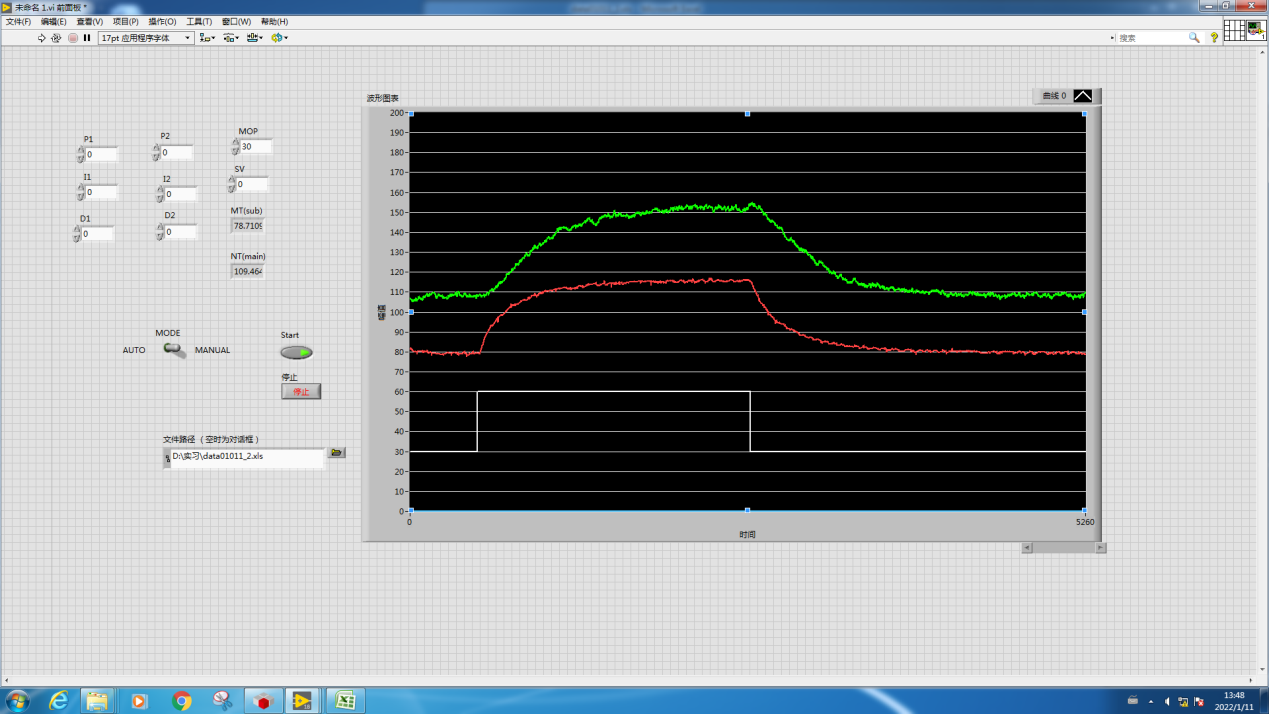


图 7-1 测试建模法图像

计算时，副对象近似为一阶环节，主副对象的串联近似为一阶滞后环节。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数值  测量值 | K | T | τ |
| 正向输入 | 1.42 | 212 | 58 |
| 负向输入 | 1.41 | 174 | 78 |
| 平均值 | 1.415 | 193 | 68 |

表 7-1 主对象参数

主对象的传递函数为：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数值  测量值 | K | T | τ |
| 正向输入 | 1.46 | 130 | -3 |
| 负向输入 | 1.44 | 142 | -10 |
| 平均值 | 1.45 | 136 | 0 |

表 7-2 副对象参数

副对象的传递函数为：

1. **整定**

整定时先整定副环再整定主环。整定主环时副环保持不变。

当副控制器P2=1.5，I2=8，D2=0时，副回路整定结果如下：

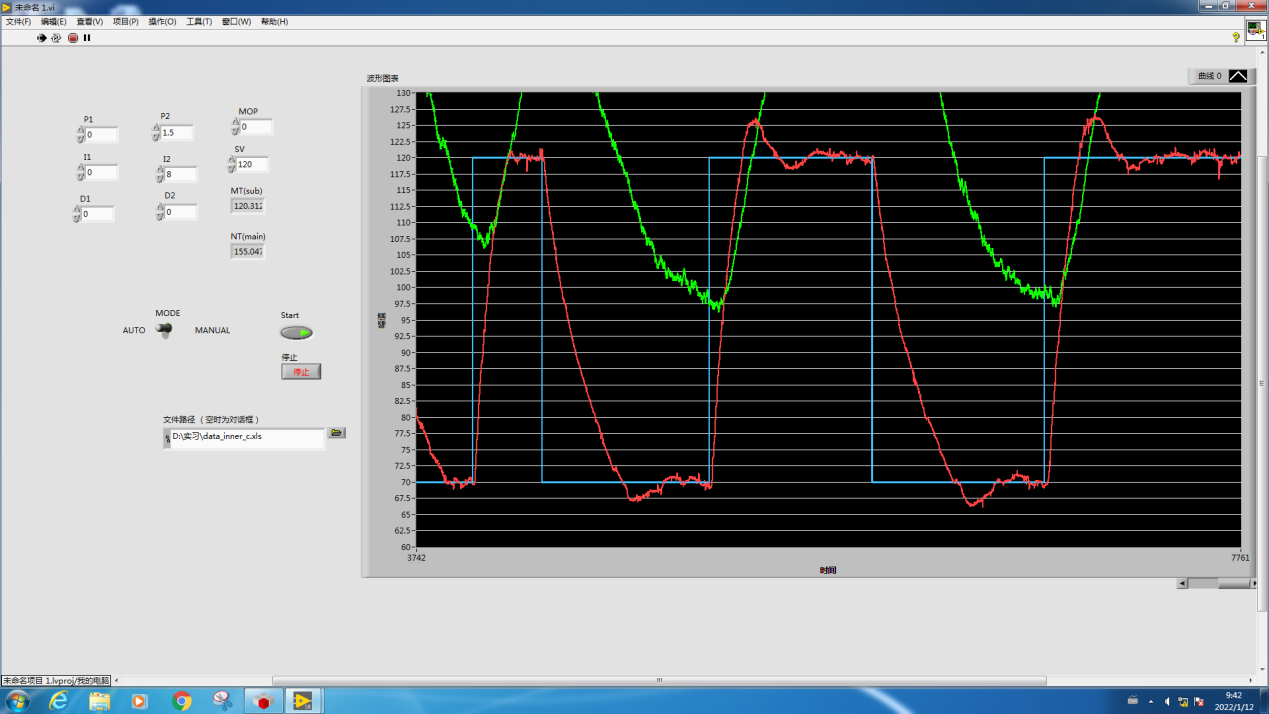


图 7-2 副回路整定图1

当副控制器P2=1.8，I2=8，D2=0时，副回路整定结果如下：

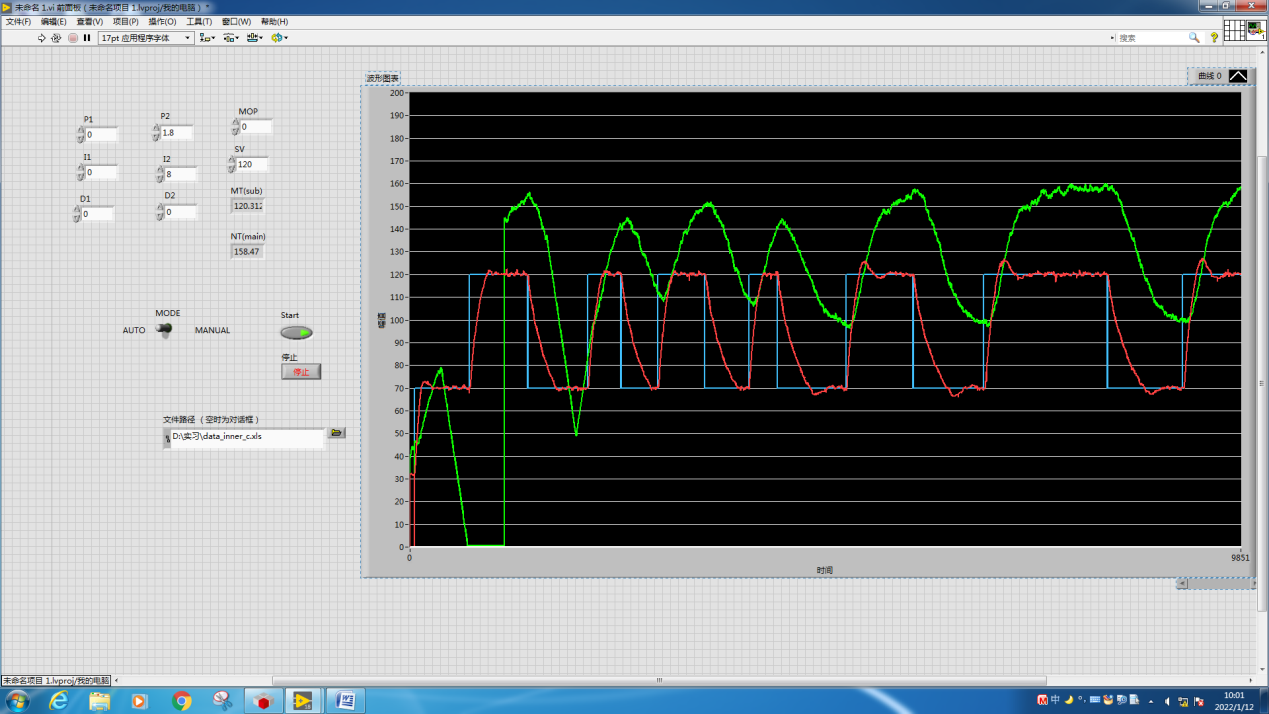


图 7-3 副回路整定图2

副回路要求较高的衰减比，能够迅速的衰减到稳定。整定副回路注重快速性。

上面两幅图的结果都能达到期望的效果，选择副控制器的PID参数P2=1.8，I2=8，D2=0作为整定结果。

整定主回路时，副控制器的PID参数保持不变，对主控制器进行整定。

当主控制器的PID参数为：P1=1,I1=30,D1=0.01，整定得到的图像为：

时图像为衰减震荡但是衰减率太低，对主控制器的PID参数进一步调节。

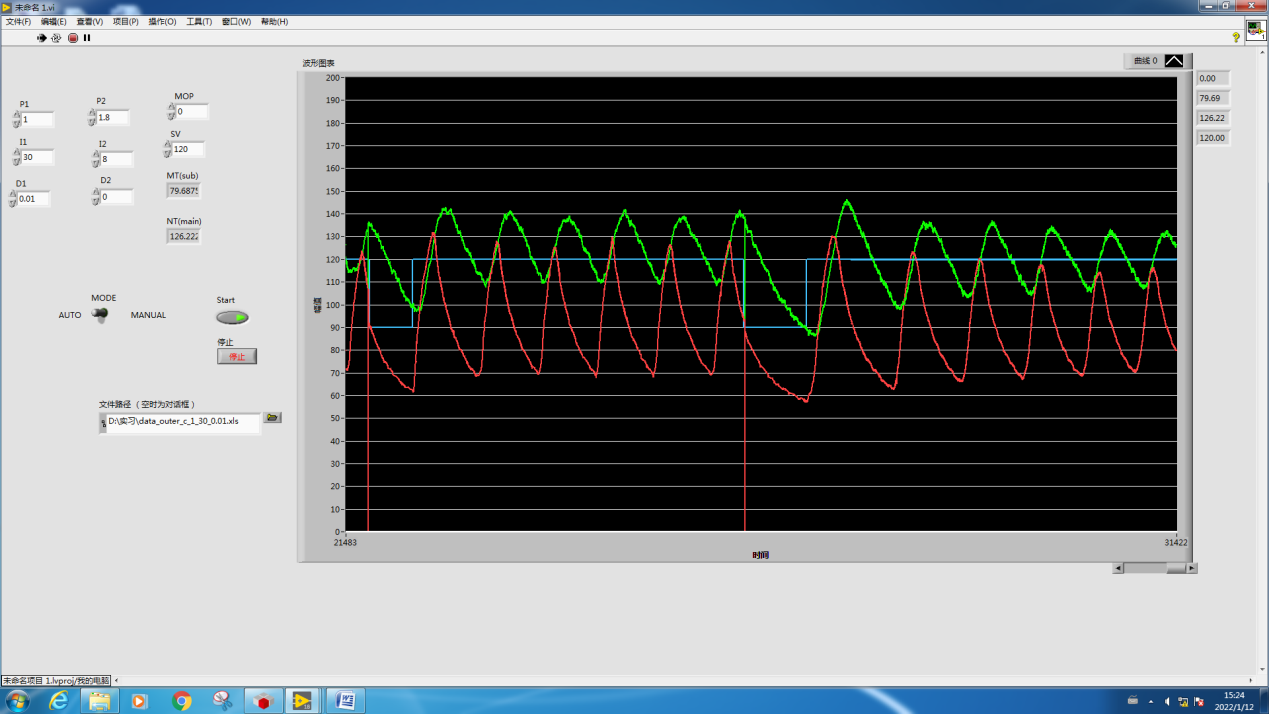


图 7-4 主回路整定结果1

主控制器的PID参数为：P1=1,I1=50,D1=0时，整定后的图像为：

此时，整定后的图像为衰减震荡最终稳定在设定的120°C上下波动。

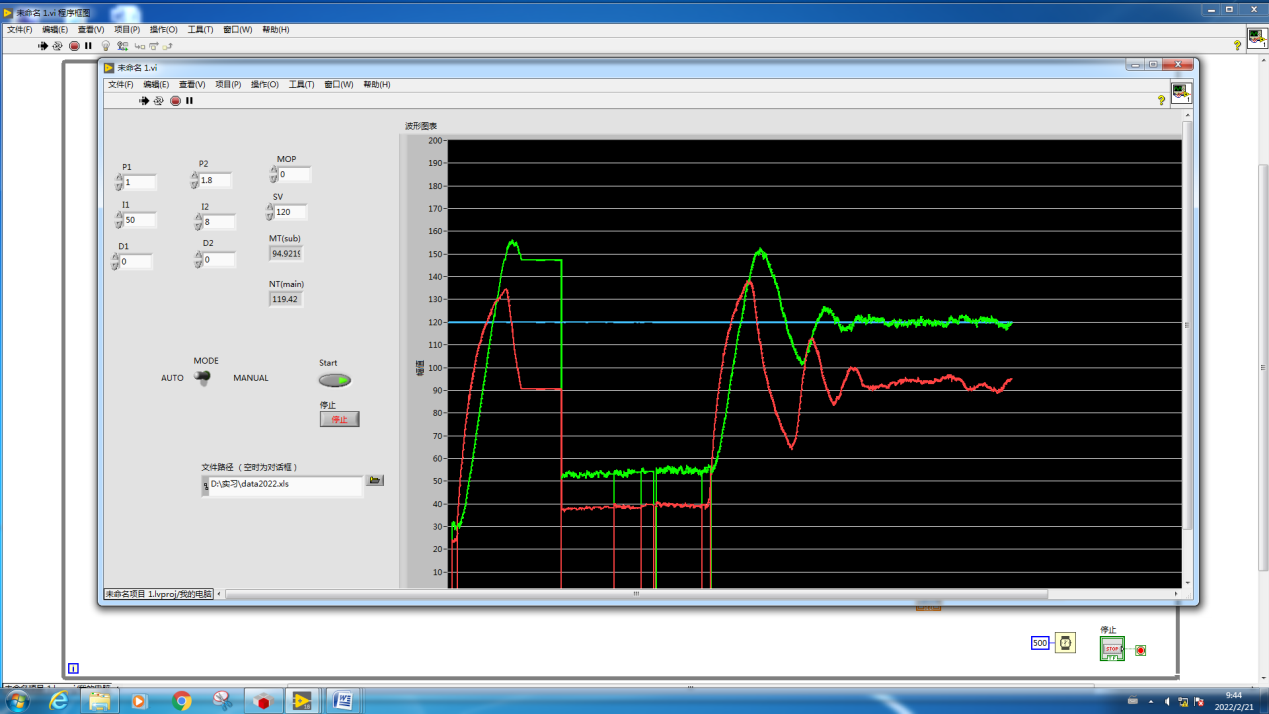


图 7-5 主回路整定结果2

1. **分析**

图 7-6 主环整定图

在调节主回路PID参数过程中，图像中的红线为副对象的温度变化，绿线为主对象的温度变化。温度从副对象传导至主对象的过程中会有一定的滞后，从图中可以看出，红线温度变化先于绿线，这种滞后，对控制过程不利。使用PID控制不能消除滞后作用。通过引入微分环节可以减小滞后环节的影响。微分作用可以在温度变化时更快的反应，使PID输出变大。但是微分环节D过大时，会使PID控制器在温度偏离设定值时输出很容易饱和，经过反复调试，选择去除主对象的微分环节。

在主、副回路均整定后得到的图像中，可以看到被控温度以衰减震荡的形式最终稳定在设定温度120度左右。由图可以看出系统温度经过几个震荡过程便稳定下来，调节时间大约15分钟，系统的快速性尚可。但是系统的超调达到了150度以上，超调量相对较大。温度最终误差范围在±3度左右，误差较大。仍有改进的空间。

1. **总结**

本次专业实习不仅仅是对过程控制系统设计与实践的深入学习，又增加了设备的选型、接线图的绘制到系统的仿真再到实际系统的建立这一过程。让我明白一个实际工程从设计到实现的复杂。

在设备选型上，因为缺乏实际工程的经验，选择了使用变频器控制加热电压这一既昂贵又没有效率的加热方式。后来在老师的指导下，选择了固态继电器这样简单有效且便宜的设备来控制加热。

在对系统进行模拟时，由于要模拟工厂的高温环境，要将温度做一定的放大。这样做同时也放大了误差，需要采用一定的滤波算法来对数据进行处理，否则不利于后面的建模和整定。一开始选用了常见的算术平均值滤波方法，通过增大采样次数来减小误差，但是尽管如此图像还是有较大的波动。后来我们又想再加入对采样到的温度变换进行限幅来减小波动。然而在进行限幅处理的时候，会导致图像在从零开始进行动态过程时也会受到限幅。最后我想到了对从零开始时进行输出的大小判断，如若输出值为零，则把实际值赋予输出值，这样不仅解决了一开始限幅遇到的问题，而且还从代码上节省了输出从零变化到实际值的时间。经过不断修改，最终达到了较好的数据滤波效果。

在最后调整主环时，一上来便遇到了关于串级控制系统的两个PID控制器的输入与输出量量纲和数据范围不对应的问题，询问老师后了解了在实际工程中，这是每个工程需要考虑和解决的问题，一般都是采用无量纲的形式给予PID控制器进行处理，本次实验中PID控制量我们也采取了无量纲的形式，范围取在0-100。最终调试阶段总是调不出衰减震荡的波形，不得不将微分作用减小，甚至最后去除了微分作用。但图形还是不能快速的衰减震荡到稳定。一组参数要等半小时才能看到成果，调了好几次最终还是不行。后来又想要采用积分分离的方法来防止积分饱和来减少温度的较大波动，但是没有成功。采用原来的参数再次调试时，经过长时间的等待却最终出现了较好的结果。

虽然最后完成了实验，但是还有许多可以改进的地方。最终采用的PID参数只是达到了要求，并没有进行进一步的调整，相信通过对PID参数的进一步调整可以达到更好的结果。本来想要加入的积分分离方法没能成功实现。在labview上设计的前面板还不够直观。在这些设计上我还缺乏实际的思想，不能自己看懂了就草草了事，应该从工程的角度改进系统。