

SMART_PLC 在箱式热风炉中的应用

郑州磨料磨具磨削研究所，韩瑞锋

摘要：在本次箱式热风炉热风控制中，使用了 SMART 系列 PLC 的 ST30 型号 CPU 作为控制器，通过对现场设备的 IO 控制以及 MODBUS 协议通信自由口通讯的控制，实现了热风循环控制的功能，并通过以自由口实现了与中控室上位机的连接，实现了对温度，风速，以及状态报警监控显示，以及灵活的修改不同的工艺配方操作，充分发挥了 ST30 型号 CPU 的系统功能，来实现对设备的控制和监控，也使我对小型机的功能有了新的认识。

关键词： S7-200 SMART、MODBUS、自由口、Visual Basic、箱式热风炉

一、项目介绍

随着超硬材料在工业现场自动化程度越来越高的大趋势，我们作为国内唯一一家从事超硬材料研究和行业的领军单位，带领国内行业发展任重而道远。由于目前国际建材业市场不断扩大致使目前超硬材料种类多生产量大，导致普通的小型固化设备远远不能满足从事树脂切割片的生产企业的生产要求，于是针对这一情况我们开发了设计了采用箱式热风循环结构，从而提高生产树脂切割片的产量。

在项目中，温度控制和数据采集交换在项目中至关重要，采用西门子 SMART 系列 PLC 做为主控制单元并且完成所有的数据采集转换。

二、工艺原理

本设备，完成对树脂切割片的固化烧结，采用电加热管作为加热源，通过温控表实时监控反馈的温度通过温控表实时上传到 PLC 进行实时比对，通过模拟量模块实时调节风机风流大小，从而吹风将热源均匀扩散。本控制系统可按照客户预设不同的升温斜率进行升温、保温、降温控制，下面参照数据曲线显示：



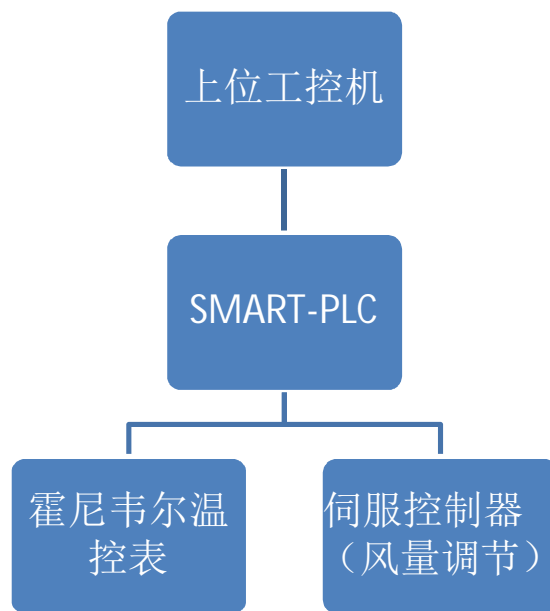
三、方案选型

1、控制方案设计

针对本系统的特点，动作控制频繁，但是点数并不多，主要采用本机通讯功能，所以采用集成了 10 点的小型 PLC 较为合适。

而较为复杂的温度控制，交给专业的温控表来实现，即节约了 PLC 资源，又降低了设计难度，同时温控表还可兼做设置和显示，而目前很多温控表都集成了 MODBUS 协议，如果 PLC 也支持 MODBUS 协议则系统具备很强的扩展性，这样就又省去了一些数字量和模拟量的 IO 点，使系统更加简洁。

另外，工厂设有中控室，可以监控生产线大部分设备，像固化烧结这类设备本身存在于一定的危险性，所以并未纳入中控范围，如果能通过采用通讯进入中控室，则更加完善了中控室系统的功能，也降低了设备对人体的危险性。



2、IO 表设计及 PLC 选型

根据现场设备，汇总需要输入点 12 个，输出点 8 个，调节动作过于频繁所以采用晶体管输出，根据 IO 点数要求，我们选择了 S7-200 SMART 系列的 ST30 型号 CPU，并根据需求设计了 IO 表。

输入			输出		
序号	名称	IO	序号	名称	IO
1	手动	I0.0	1	故障报警灯	Q0.0
2	自动	I0.1	2	运行状态灯	Q0.1
3	急停	I0.2	3	保持状态灯	Q0.2
4	程序继续	I0.3	4	风机运行灯	Q0.3
5	程序保持	I0.4	5	加热运行灯	Q0.4
6	程序跳步	I0.5	6		Q0.5
7	风机启动	I0.6	7		Q0.6
8	风机停止	I0.7	8		Q0.7
9	加热	I1.0	9	风机启动	Q1.0

10	停热	I1.1	10	加热调节	Q1.1
11	风机过载	I1.2	11	加热开关	Q1.2
12	炉内报警	I1.3	12		Q1.2

3、其它主要电气设备选型

另外选择了其他主要设备如下：

序号	名称	型号	数量	备注
1	通讯模块	6ES7 288-5CMQ1-0AA0	1	支持 MODBUS 通信
2	模拟量模块	6ES7 288-3AQ02-0AA0	1	模拟量调节
3	温控表	霍尼韦尔 DC1040	6	支持 MODBUS 通信
4	伺服控制器	霍尼韦尔 CN75	2	风流调节

4、确定通讯方案

1) 上位机自由口通信方案

本工厂使用的上位机为自己开发的 Visual Basic 上位机组态软件，自定义通讯协议，协议设置格式如下：

波特率：9600

通讯协议：自由口

校验：8N1

2)、MODBUS 通讯方案

①霍尼韦尔 DC1040 温控表采用标准 Modbus 协议，能够方便地接入系统。

从站地址	功能码	数据	故障检出
------	-----	----	------

其中功能码如下：

功能码（十进制）	功能
40139	读取各分站温度值
40001	向各分站写温度设定值

②霍尼韦尔温控表参数设置

NO.	名称	设置内容
LEVEL3-08	协议选择	RTU
LEVEL3-12	校验方式	8N2
LEVEL3-16	站地址	1—6
LEVEL3-17	波特率	9600

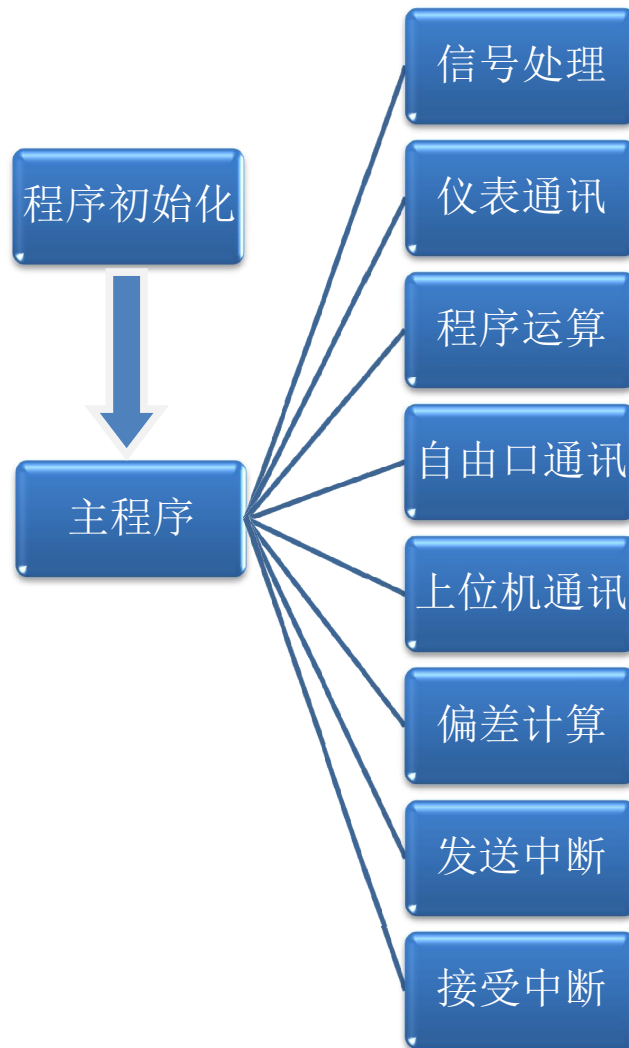
四、编程与调试

1、10 控制程序编程与调试

由于程序中工艺段数较多需要处理的数据量较大，故程序中大量使用子程序用以提高增强程序的运算。

主要控制动作很简单，动作只有风机启动启动停止。该控制主要用到模拟量控制模块，该模块主要调节调节伺服阀的开度大小，根据仪表反馈回的开度值，进行自动调节。程序内部做了一些程序算法用以调节。

程序的主体轮廓参看下图：



2、上位机通信程序编程与调试

上位机首先与 PLC 通过西门子的 232 转 485 通讯线建立数据连接,我们采用的是自由口通讯所以应将拨码开关 2 调到 on 位, 然后连接上位机与 PLC 的 COM0 口。

应用自由口通信首先要把通信口定义为自由口模式,同时设置相应的通信波特率和上述通信格式。用户程序通过特殊存储器 SMB30 (对端口 0 即 CPU 本体集成 RS485 口)、SMB130 (对端口 1 即通信信号板) 控制通信口的工作模式。

CPU 通信口工作在自由口模式时, 通信口就不支持其他通信协议 (比如 PPI)。

通信口的工作模式, 是可以在运行过程中由用户程序重复定义的。自由口通信的核心指令是发送 (XMT) 和接收 (RCV) 指令。自由口通信常用的中断有“接收指令结束中断”、“发送指令结束中断”, 以及字符接收中断。用户程序不能直接控制通信芯片而必须通过操作系

统。用户程序使用**通信数据缓冲区**和**特殊存储器**与操作系统交换相关的信息。

XMT 和 RCV 指令的数据缓冲区类似，起始字节为需要发送的或接收的字符个数，随后是数据字节本身。如果接收的消息中包括了起始或结束字符，则它们也算数据字节。

调用 XMT 和 RCV 指令时只需要指定通信口和数据缓冲区的起始字节地址。

XMT 和 RCV 指令与网络上通信对象的“地址”无关，而仅对本地的通信端口操作。如果网络上有多个设备，消息中必然包含地址信息；这些包含地址信息的信息才是 XMT 和 RCV 指令的处理对象。

由于 S7-200 SMART 的通信端口是半双工 RS-485 芯片，XMT 指令和 RCV 指令不能同时有效。

上位机可以方便的将编程工艺曲线存储进上位机里，客户还可以调出之前预设工艺数据以适应不同的生产工艺，同时在运行过程中能够很直观实时监控运行曲线，运行结束后还可以调出数据记录，便于分析问题。

1) 通讯命令数据内容:

上位机指令格式 (ASCII 码): 共 210 个字符

“:” + “ W/R” + “ X” + “ M” + “数据内容” + BCC + CR

1、起始字符

“:” 3AH

2、指令类型

“ W” 57H 写数据, “ R” 52H 读数据

3、指令代码 X

“ 1” 31H 写数据 (条号、每段的起始时间和停止时间)

“ 2” 32H 写数据 (温度、偏差值、阈值)

“ 3” 33H 读数据 (各温度测量值、状态值、报警位)

4、读/写数据字节数 M, 占 2 个字节。

M= 202 写条号, 每段起始和停止时间, 占 202 个字节

M= 150 写温度、偏差值、阈值, 占 150 个字节

M= 48 读温度测量值和状态值, 占 48 个字节

5、数据内容

共 202 个字节, 前 M 字节有效。

6、校验格式

BCC 块校验，“数据内容”异或值，占 2 个字节。

7、结束字符

“ CR”0DH

8、“数据内容”的传输格式：

(st1_H 为起始时间 1 高字节，st1_L 为起始时间 1 低字节，sp1_H 为停止时间 1 高字节，sp1_L 为停止时间 1 低字节，Temp1 为温度 1，Dep1 为偏差 1，Exh1 为阈值 1)

X=“ 31H” 时，传输格式为：

条号+ st1_H + st1_L + st2_H + st2_L +.....+ st25_H + st25_L + sp1_H + sp1_L + sp2_H + sp2_L
+.....+ sp25_H +sp25_L

X=“ 32H” 时，传输格式为

Temp1+ Temp2 ++Temp25+ Dep1 + Dep2 +.....+ Dep25+ Exh1 + Exh2 ++ Exh25

X=“ 33H” 时，传输格式为

202 个字节全部用 00H 补充

下位机 (PLC) 应答指令格式 (ASCII)

“:” + “ SM” + “数据内容” + BCC + CR

1、起始字符

“:” 3AH

2、状态信息 SM

SM=“ 1”，31H 条号、每段起始时间和停止时间写入正确

SM=“ 2”，32H 温度、偏差值、阈值写入正确

SM=“ 3”，33H 温度测量值、状态位、报警位读取正确

SM=“ 4”，34H BCC 校验错误

3、数据内容

(Ztim_H 为总运行时间高字节，Ztim_L 为总运行时间低字节，Dtim_H 为段运行时间高字节，Dtim_L 为段运行时间低字节)

格式：条号+段号+Ztim_H + Ztim_L+ Dtim_H + Dtim_L + 烤箱温度 SV + 烤箱温度平均 PV + 温度 PV1 + 温度 PV2+.....+温度 PV12 + 砂轮温度 PV + 状态字节 1 + 状态字节 2 + 状态字节

3 。ASCII 码共计占 48 个字节。

注：状态字节每位定义由上位机监控界面确定。

4、校验格式

BCC 块校验，“数据内容”异或，占 2 个字节。

5、结束字符

“ \n” 0AH

3) 通信程序编写

①、端口 0 初始化:

SMB30: 偶校验、8 位数据位、9600bps、自由口协议

SMB87: 不使用 SMB88 检测起始信息、不使用 SMB89 检测结束信息、使用 SMW90 检测空闲状态、使用内部字符定时器、当执行 SMW92 时终止接收、忽略终端条件检测起始信息

SMW90: 空闲线时间>5ms 接收信息为开始

SMW92: 定时器>5ms 停止接收信息

SMW94: 最大字节 50

②、通信指令:

通讯指令使用 XMT 和 RCV 指令进行数据的发送和接收。

每秒钟使用指令数据读取一次温度的设定值和测量值

当有按下风机启动、加热启动时，发送状态启动指令，当修改设定值时，根据当前的运行状态输出一次指令，并发送新的设定值和状态标志位。

3、温控表通信程序编程与调试

温控表采用标准的 Modbus 通讯协议以实现数据的交换，首先将 COM1 端口装入 PLC 扩展口，用两芯双绞屏蔽线将温控表和 PLC 通讯接口连接起来。

. 调用 Modbus RTU 主站初始化和控制子程序

使用 SM0.0 调用 MBUS_CTRL 完成主站的初始化，并启动其功能控制:

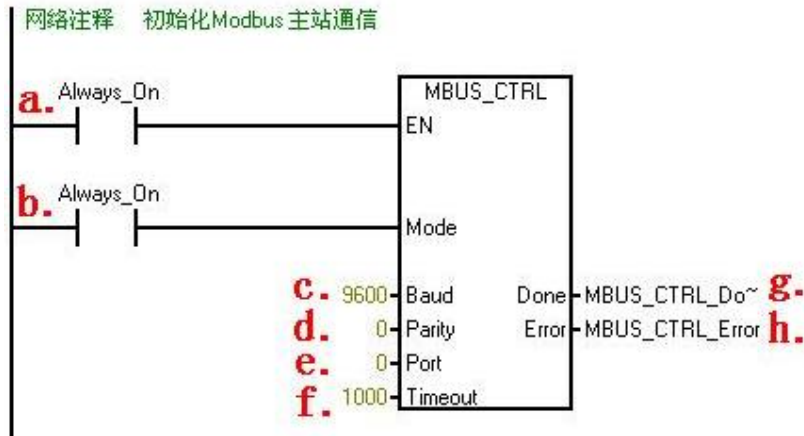


图 2. 用 SMO.0 调用 Modbus RTU 主站初始化与控制子程序

各参数意义如下:

- a. EN** 使能: 必须保证每一扫描周期都被使能 (使用 SMO.0)
- b. Mode** 模式: 为 1 时, 使能 Modbus 协议功能; 为 0 时恢复为系统 PPI 协议
- c. Baud** 波特率: 支持的通讯波特率为 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200。
- d. Parity** 校验: 校验方式选择
0=无校验
1=奇较验
2=偶较验
- e. Port** 端口号: 0 = CPU 集成的 RS 485 通讯口 ; 1 = 可选 CM 01 信号板 。
- f. Timeout** 超时: 主站等待从站响应的的时间, 以毫秒为单位, 典型的设置值为 1000 毫秒 (1 秒), 允许设置的范围为 1 - 32767。
注意: 这个值必须设置足够大以保证从站有时间响应。
- g. Done** 完成位: 初始化完成, 此位会自动置 1。可以用该位启动 MBUS_MSG 读写操作 (见例程)
- h. Error** 初始化错误代码 (只有在 Done 位为 1 时有效):
0= 无错误
1= 校验选择非法
2= 波特率选择非法
3= 超时无效
4= 模式选择非法
9= 端口无效
10= 信号板端口 1 缺失或未组态

2. 调用 Modbus RTU 主站读写子程序 MBUS_MSG, 发送一个 Modbus 请求;

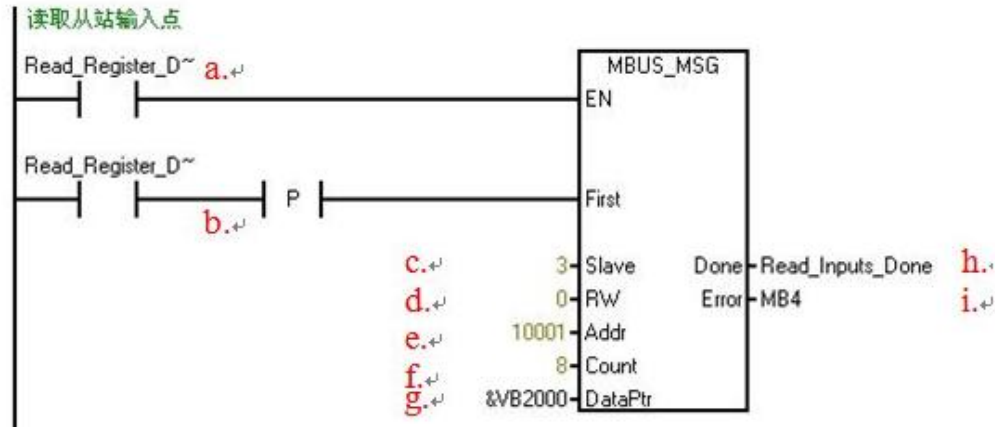


图 3. 调用 Modbus RTU 主站读写子程序

各参数意义如下:

- a. EN** 使能: 同一时刻只能有一个读写功能 (即 MBUS_MSG) 使能
注意: 建议每一个读写功能 (即 MBUS_MSG) 都用上一个 MBUS_MSG 指令的 Done 完成位来激活, 以保证所有读写指令循环进行 (见例程)。
- b. First** 读写请求位: 每一个新的读写请求必须使用**脉冲**触发
- c. Slave** 从站地址: 可选择的范围 1 - 247
- d. RW** 读写请求: 0 = 读, 1 = 写
注意:
 1. 开关量输出和保持寄存器支持读和写功能
 2. 开关量输入和模拟量输入只支持读功能
- e. Addr** 读写从站的数据地址: 选择读写的数据类型
 00001 至 0xxxx - 开关量输出
 10001 至 1xxxx - 开关量输入
 30001 至 3xxxx - 模拟量输入
 40001 至 4xxxx - 保持寄存器
- f. Count** 数据个数: 通讯的数据个数 (位或字的个数)

数

注意： Modbus 主站可读/写的最大数据量为 120 个字（是指每一个 MBUS_MSG 指令）

- g. DataPtr 数据指针： 1. 如果是读指令，读回的数据放到这个数据区中
2. 如果是写指令，要写出的数据放到这个数据区中
- h. Done 完成位 读写功能完成位
- i. Error 错误代码： 只有在 Done 位为 1 时，错误代码才有效
- 0 = 无错误
 - 1 = 响应校验错误
 - 2 = 未用
 - 3 = 接收超时（从站无响应）
 - 4 = 请求参数错误（slave address, Modbus address, count, RW）
 - 5 = Modbus/自由口未使能
 - 6 = Modbus 正在忙于其它请求
 - 7 = 响应错误（响应不是请求的操作）
 - 8 = 响应 CRC 校验和错误
 -
 - 101 = 从站不支持请求的功能
 - 102 = 从站不支持数据地址
 - 103 = 从站不支持此种数据类型
 - 104 = 从站设备故障
 - 105 = 从站接受了信息，但是响应被延迟
 - 106 = 从站忙，拒绝了该信息
 - 107 = 从站拒绝了信息
 - 108 = 从站存储器奇偶错误

常见的错误：

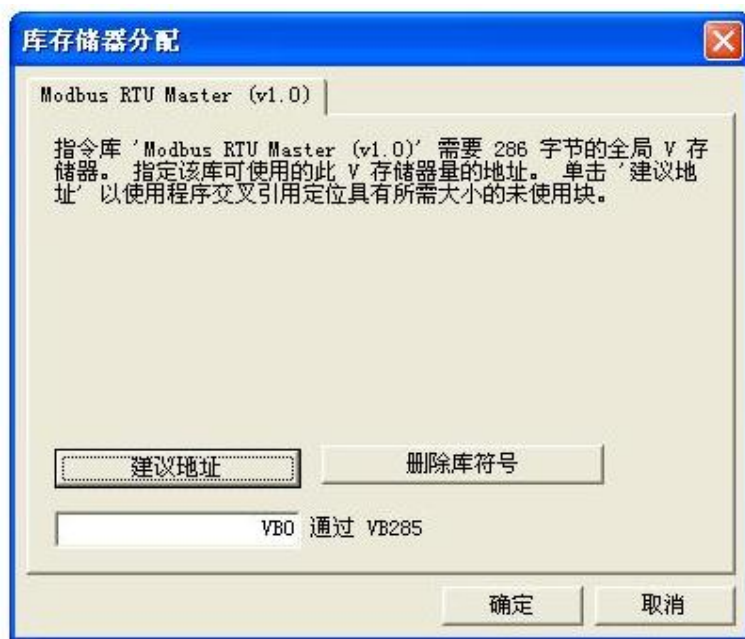
- 如果多个 MBUS_MSG 指令同时使能会造成 6 号错误
- 从站 delay 参数设的时间过长会造成主站 3 号错误
- 从站掉电或不运行，网络故障都会造成主站 3 号错误

3. 在 CPU 的 V 数据区中为库指令分配存储区（Library Memory）

Modbus Master 指令库需要一个 286 个字节的全局 V 存储区。

调用 STEP 7 - Micro/WIN SMART Instruction Library（指令库）需要分配库指令数据区（Library Memory）。库指令数据区是相应库的子程序和中断程序所要用的变量存储空间。

如果在编程时不分配库指令数据区，编译时会产生许多相同的错误。



五、项目心得体会

本次使用的 ST30 型号 CPU 的 PLC 进行系统设计和编程，整体感觉使用性能良好，很适合在系统点数要求不高的场合使用。我觉得本次使用最大的优点是在界面设计上更加人性化，对于编程来说就很方便，能够在编程中给予提示，不用经常去翻手册，大大节省了编程时间，有效的提高了工作效率。与之前的控制系统相比，虽然控制点数上略有减少，但是本体集成的串口为通讯总线的建立提供了非常便利的条件，使得许多较为复杂的设备可以简单地与 PLC 连接，也使得 PLC 本身具有了可以扩展的特性。同时，最值得提出的是以太网口的集成，不但提高了通信的速度，而且使 SMART 系列 PLC 不仅能连接支持以太网口的 HMI，还能与上位机连接，这在小型机领域，确立了其在操作和监控方面具备了不同寻常的表现。最后希望西门子（中国）能够很好推广，热卖！！

作者简介

韩瑞锋（1983-），男，电气工程及其自动化专业，长期从事自动化系统的设计开发工作。

参考文献

- 【1】 《Honeywell DC1000 系列通用调节器说明书》，美国霍尼韦尔
- 【2】 《S7-200 SMART PLC 串口通信说明》，西门子（中国）有限公司
- 【3】 《S7-200_SMART 系统手册》，西门子（中国）有限公司