



Value & Technology

可编程序控制器 **SJ-Ether 系列**
PID功能技术资料
[第二版]

捷太格特电子(无锡)有限公司
JTEKT ELECTRONICS (WUXI) CO.,LTD.

前言

承蒙采用本公司的SJ-Ether系列可编程序控制器，在此表示衷心感谢！

本手册较为详细地介绍了SJ-Ether系列PLC的PID功能，包括PID回路特点、PID回路参数设置、PID算法、PID回路整定、过程报警等，为用户应用PID控制提供了必要的信息。

除本技术资料外，与 SJ-Ether 系列 PLC 有关的技术资料还有以下，需要时请一并参考阅读。

- 《SJ-Ether 系列 PLC 用户手册》
- 《S 系列 PLC 编程手册》
- 《KPP Soft 工具软件入门手册》
- 《SJ-Ether 系列 PLC 高速计数/脉冲输出功能技术资料》
- 《SJ-Ether 系列 PLC EthernetIP 通信功能技术资料》

如果你在阅读本手册资料或使用 SJ-Ether 系列 PLC 时有什么疑问，或你需要另外的信息时，请与本公司本部或驻各地办事处联系，以便尽快得到服务。

由于产品的改进等原因，本资料所刊内容会与实际的产品有些差别，请注意！

本公司保留对包括此资料在内的所有信息的专利权！

如果你有有关本资料的情况需要与我们联系，请首先确定手册的版本号！

资料名称：《SJ-Ether系列PID功能技术资料》

资料编号	编制日期	内容说明
KEW-M3727A	2018年4月	原稿第一版
JELWX-M3727B	2024年7月	公司名称变更

目录

第一节	SJ-Ether PID 控制	4
一、	主要特点	4
二、	PID 控制基础	6
第二节	PID 控制参数设定	8
一、	回路控制表和回路数	8
二、	PID 错误标志	8
三、	设置控制表的空间大小和位置	9
四、	控制回路参数表字定义	10
五、	PID 控制模式设定 1 的位说明 (Addr+00)	11
六、	PID 控制模式设定 2 的位说明 (Addr+01)	12
七、	模式/报警监控字 (Addr+06)	13
八、	上升/保持表标志 (Addr+33)	13
九、	上升/保持表地址 (Addr+34)	14
十、	上升/保持表设置错误标志 (Addr+35)	14
第三节	控制采样周期和时间表	15
一、	控制采样周期 Addr+07	15
二、	选择最佳的采样周期	15
三、	确定合适的采样周期 (Addr+07)	16
四、	采样周期编程	16
五、	CPU 扫描对 PID 回路的影响	17
第四节	过程控制成功的十个步骤	19
第五节	基本回路动作	21
一、	数据地址	21
二、	数据传送	21
三、	使用滤波器选项的 PV 自动传输功能	21
四、	回路控制模式	22
五、	控制模式变更方法	23
六、	使用操作面板控制 PID 模式	24
七、	PLC 模式对控制模式的影响	24
八、	回路控制模式的覆盖	25
九、	无扰动切换	25
第六节	PID 控制的数据构成	27
一、	控制参数的数据格式	27
二、	选择有符号或无符号	27
三、	数据偏移	28
四、	设定值 (SP) 的限幅	28
五、	远程设定值 (SP) 地址	28
六、	过程变量 (PV) 的设置	29
七、	控制输出设置	30
八、	误差项的设置	30
第七节	PID 算法	32
一、	位置算法	32

二、速度算法	33
三、正方向和反方向回路	33
四、比例、积分、微分项	34
五、部分使用 PID 控制	36
六、微分增益限幅	36
七、偏差项	36
八、偏差冻结（积分分离）	37
第八节 PID 回路整定方法	38
一、开环测试	38
二、手动整定方法	38
三、自整定方法	39
四、串级回路的整定	43
五、在程序中加入模拟量滤波	44
第九节 前馈控制	46
第十节 时间-比例控制	48
第十一节 串级控制	50
一、简介	50
二、SJ-Ether CPU 的串级回路	51
第十二节 过程报警	52
一、PV 绝对值报警	52
二、PV 偏离报警	53
三、PV 变化率报警	54
四、PV 报警滞后	54
五、报警设置出错	55
六、回路计算上溢/下溢错误	55
第十三节 上升/保持发生器	56
一、简介	56
二、上升/保持表	57
三、上升/保持表的标志	58
四、上升/保持程序控制有效	59
五、上升/保持控制	59
六、上升/保持程序控制的监控	60
七、上升/保持编程错误	60
八、测试上升/保持动作	60
第十四节 利用 KPPsoft 软件设置 PID 参数	61
一、打开 PID 设定窗口	61
二、设置参数	62
三、PID 设置及编程举例	66
四、PID 运行监视	72
第十五节 故障检修提示	73
附录 1 PID 回路术语汇总	75

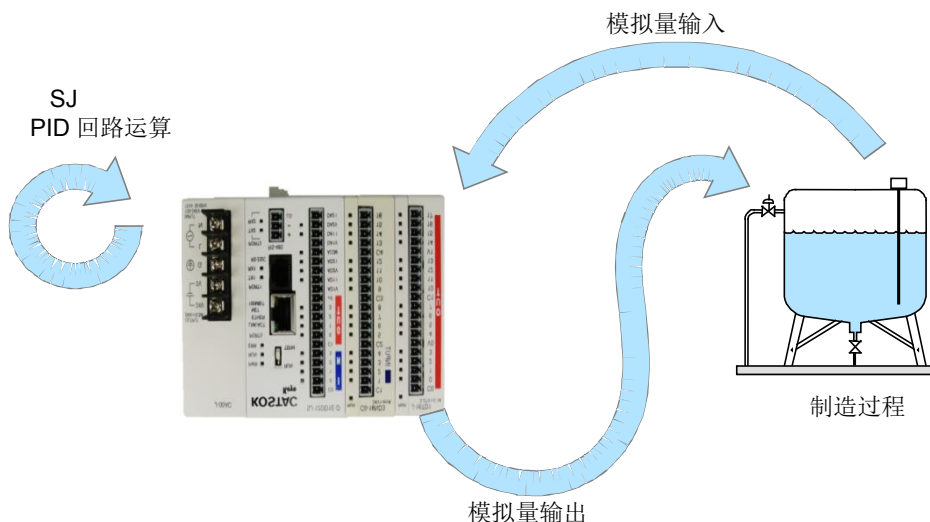
第一节 SJ-Ether PID控制

一、主要特点

SJ-Ether的过程回路控制，提供了面向多种应用需要的功能。主要特点如下：

- 最多16个回路，采样速率可单独编程
- 具有手动/自动/串级控制功能
- 具有两种无扰动转换方式
- 丰富的报警功能
- 最大16段上升/保持程序控制
- 自整定

SJ-Ether CPU除了能够执行梯形图程序以外，还具有过程控制(PID)功能, 最多可选择组态16个回路。传感器和执行机构的线缆可以直接连接到SJ-Ether。每个PID控制回路的过程变量、增益值、报警级别等都存放在CPU内的设置参数表内。每次扫描时，SJ-Ether CPU读取过程变量(PV)的输入，然后在每个PLC扫描周期内进行PID回路的计算，并刷新控制输出值。控制回路利用比例(P)–积分(I)–微分(D)算法来产生控制输出。本章描述了PID回路是如何运行，为构成回路如何进行调整。



可以使用编程软件KPPsoft设置SJ-Ether的PID控制。在KPPsoft软件中，使用对话框逐一对PID回路进行设定。完成设置后，可以用软件的PID 趋势图，调整各PID回路控制。PID的设置保存在数据寄存器中，清除寄存器信息时，PID设置也会消除。控制参数可以保存在磁盘上以供日后调用。



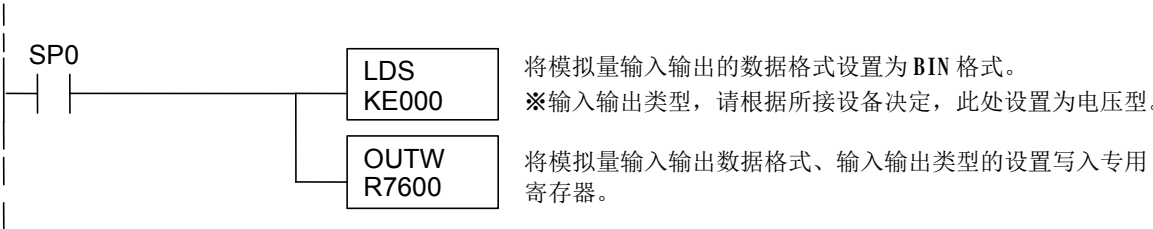
注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

PID回路性能	规格
回路数	最多16个回路
占用CPU存储器	每回路32字(R存储区)，使用上升/保持程序控制时，则为64字。
PID算法	位置型和速度型
控制方向	可选正向控制或反向控制
误差曲线	可选择为线性、误差的平方
采样时间(PID的运算时间间隔)	0.05~99.99s(用户可编程)
最小采样时间	0.05s(1~16回路)
控制模式	自动、手动或串级控制
上升/保持程序控制	每个回路最多8步上升/保持(16段)
PV曲线	选择标准线性或开方(流量仪表输入)
设定值上下限	规定设定值的最小和最大值
测量值上下限	规定测量值的最小和最大值
增益(P)	规定增益为 0.01~99.99
重设时间(Ti)	规定重设时间为0.1~999.8(以秒或分为单位)
微分时间(Td)	规定微分时间为0.01~99.99s
微分增益限幅	规定微分增益限幅为1~20
无扰动切换 I	从手动模式切换到自动模式时, 自动初始化误差和设定值
无扰动切换 II	从手动模式切换到自动模式时, 自动设置误差为控制输出
阶跃误差	设定值变化大时, 提供比例误差调整
重置饱和	当控制输出达到0%或100%时，禁止积分操作(防止积分动作引起的大幅过冲)
误差死区	指定误差项(SP-PV)的变动范围(+/-)，在此范围内，误差当作0计算进行控制。

报警功能	规格
死区	规定所有报警的报警死区是0.1%~5%
PV值报警	在低-低、低、高和高-高条件时，设置PV报警的设定值
PV偏差报警	规定PV值(测量值)偏离SP值(设定值)的两个报警范围
PV变化率报警	检测PV值是否超过了用户规定的变化率

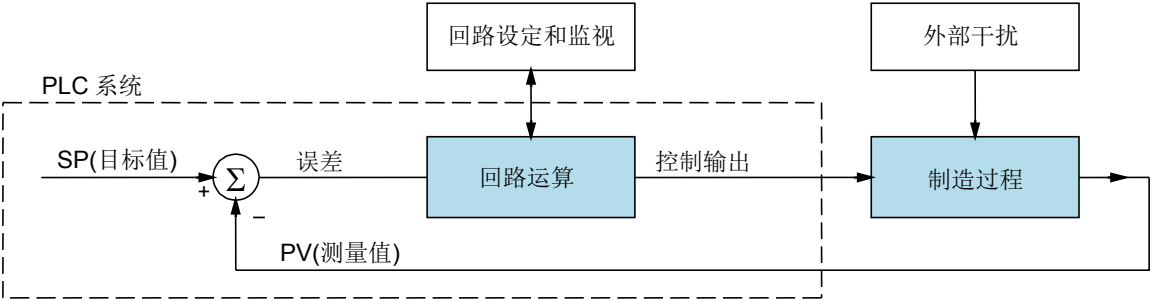


注意：使用SJ-Ether CPU的模拟量输入输出时，在专用寄存器R7640(模拟量输入输出设置寄存器)将数据格式设置为“BIN”格式。
软件KPP的PID设定/PID监视设置为“BCD”格式。



二、PID 控制基础

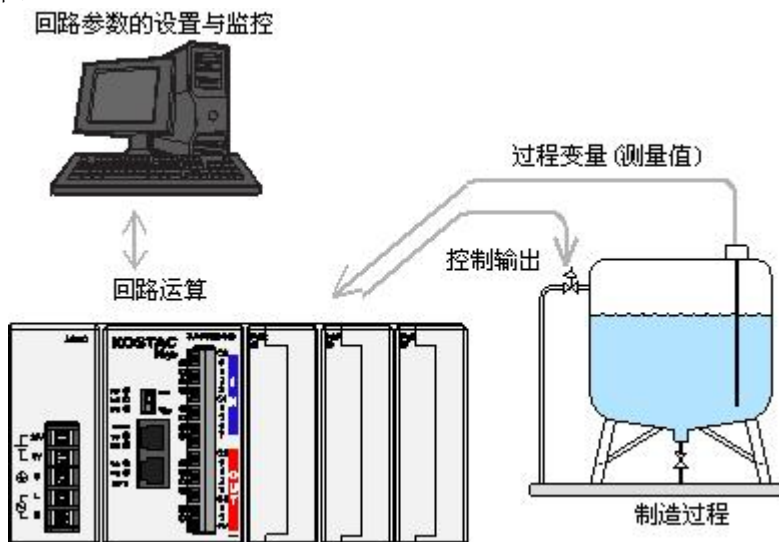
PID控制的主要组成部分如下面的方块图所示。从PLC到制造过程再回到PLC的这条路径，就是“闭环回路控制”。



制造过程	对原料施加一系列的处理，此过程可能包括材料的物理变化和/或化学变化。这种变化会使材料更利于其特定的用途，并最终有利于其成品的完成。
过程变量(测量值)	用几种类型的传感器对原材料的一些物理特性进行测量。例如，如果制造过程使用加热炉，要控制的就是温度，温度就是一个过程变量。
设定值(目标值)	过程变量最终目标值，能够生产出最佳产品。设备操作者知道这个值后，手动设置或对PLC编程，以便自动运行使用。
外部干扰	控制系统试图消除不可预知的外部因素的干扰。例如，如果燃料的输入是恒定的，那么加热炉在天热时要比在天冷时更热。加热炉控制系统必须抵销这种外部影响，在任何季节里都维持一个恒定的炉温。因此，天气（无法预料的）是此过程的干扰源之一。
误差	过程变量(测量值)与设定值(目标值)之间的差异称为误差。当过程变量(测量值)等于设定值(目标值)时，误差等于零。一个运行良好的PID控制，能够使误差幅度较小。
PID控制运算	应用程序实时对误差项的进行数学运算，产生合适的控制输出命令，使误差量最小化。
控制输出	PID控制运算的结果输出到控制过程。（如同加热炉的加热一样）
设置PID控制参数	选择实行PID控制并优化性能。PID控制运算功能使用设置的参数，实时调整增益、偏差量等。
PID控制监控	使操作者可以观察PID控制的状态和性能。可以与设置PID控制参数一起使用，优化PID控制性能(使误差项最小)。

下图显示了PID控制的基本构成。以控制容器中的液体为例，使用传感器探头测量过程变量，过程变量可以是压力、温度或其它变量。传感器信号通过一个变送器被放大，通过导线以模拟量的方式传送到PLC的输入模块。

PLC从模拟量输入模块上读取PV值(测量值)，CPU进行PID控制运算，运算的结果为模拟量值。此信号送到制造过程中的加热器、阀、泵等执行机构，一段时间后，液体的变化可以被传感器探头测量到后，过程变量也相应地发生变化，下一个PID控制开始，PID控制运算循环就是以这种方式连续工作。



第二节 PID控制参数设定

一、回路控制表和回路数

SJ-Ether PLC使用寄存器表进行PID控制。因此在KPP中，梯形图程序中没有PID指令，而是CPU从保留的寄存器区读取配置参数。如下表所示，存放控制参数的寄存器区的起始地址存放在R7640中，PID运算的回路数存放在R7641。当通过程序写入的R7640和R7641设置信息不正确时，在R7642中写入设置错误信息。

地址	参数设置	数据类型	范围	读/写
R7640	控制参数表起始地址	八进制	R00400~R00640 R01400~R07340 R10000~R25740	写
R7641	回路数	BCD	0~16	写
R7642	错误标志	二进制	0或1	读

如果回路数是“0”，执行梯形图程序时，不实行PID控制。PID控制器从1开始，按升序执行。例如，不能跳过回路2和3，仅执行回路1和4。PID控制器执行的回路数，在R7641中设置。

二、PID 错误标志

在R7640和R7641中设置的信息不准确，CPU会将错误信息写入R7642。当CPU模式由STOP模式转换为RUN模式时，如果设置有错误，在R7642中相应的位置ON。当使用KPP的PID控制设置对话框时，自动检查设置范围，不会出现设置错误。由于可以使用其他方式写入设置参数，如使用梯形图程序，错误标志寄存器还是有必要的。下表是R7642中错误标志的信息。



位	错误说明 (0为无错误，1为有错误)
0	起始地址 (R7640中) 被设置为寄存器范围下限。
1	起始地址 (R7640中) 被设置为寄存器范围上限, 或超出回路表边界R27777。
2	选择的回路数 (R7641中) 大于16。
3	PID控制表超出R7377的边界。PID表每个回路使用32个字，请根据回路数设置在可用范围内。

作为一种简单的检查方法，如果CPU处于Run模式，并且R7642=0000，就是没有发生编程错误。

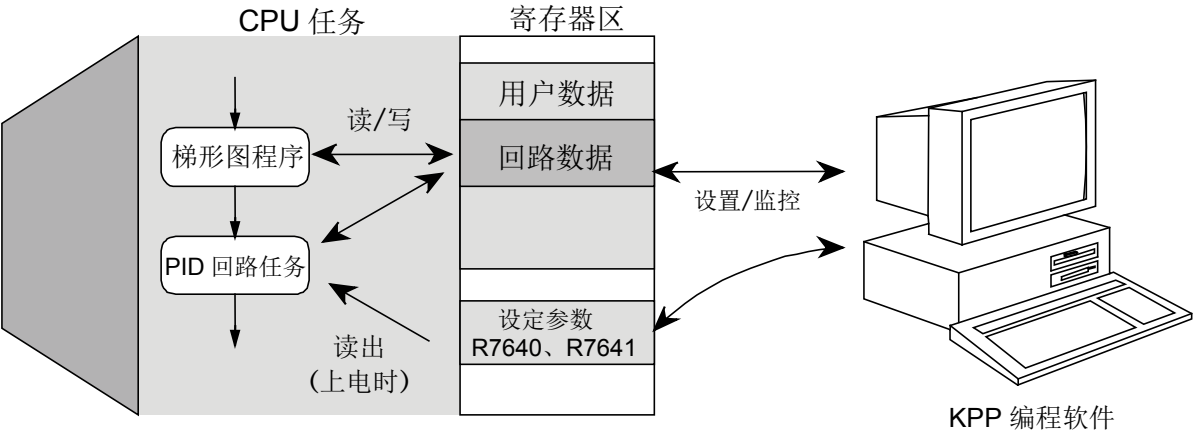
用梯形图程序直接写入PID控制表(起始地址、回路数等)，请注意设定范围。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

三、设置控制表的空间大小和位置

如下所示，当CPU从STOP模式切换到RUN模式时，CPU读取控制参数。此时，CPU就获取了控制表的位置和回路数。在梯形图程序执行期间，PID控制使用回路数据执行计算、生成报警等。每次回路运算时，CPU将读出或写入一些控制表的参数。



控制参数表包含了在R7641中设置的回路数的数据。每个回路设置在控制参数表中占用32字(八进制0~37)。

例如，假设应用程序使用了4个回路，并且选择了R2000作为起始地址。那么，回路1的控制参数占用R2000-R2037，回路2占用R2040-R2077，回路3占用R2100-R2137，回路4占用R2140-R2177

寄存器		用户数据
	↑ R 2000	回路#1
	↓ R 2037	32 字
	↑ R 2040	回路#2
	↓ R 2077	32 字
	⋮	回路#3
		32 字
		回路#4
		32 字

四、控制回路参数表字定义

下表总结了每个回路的控制相关参数。地址偏差量是八进制的，便于在控制回路参数表中找到参数。例如，如果表的起始地址为R2000，那么重设(积分)项的地址为Addr+11，即R2011。计算地址时，请不要使用字号(第一列中)。

字号	地址+偏移量	说明	类型	是否立即可读
1	Addr+0	PID控制模式设置1	位	是
2	Addr+1	PID控制模式设置2	位	是
3	Addr+2	设定值(SP)	字/二进制	是
4	Addr+3	过程变量(PV)	字/二进制	是
5	Addr+4	偏差项	字/二进制	是
6	Addr+5	控制输出	字/二进制	是
7	Addr+6	控制模式和报警状态	位	—
8	Addr+7	采样周期设定	字/BCD	是
9	Addr+10	增益(比例)设定	字/BCD	是
10	Addr+11	重设(积分)时间设定	字/BCD	是
11	Addr+12	变化率(微分)时间设定	字/BCD	是
12	Addr+13	PV值低-低报警	字/二进制	否*
13	Addr+14	PV值低报警	字/二进制	否*
14	Addr+15	PV值高报警	字/二进制	否*
15	Addr+16	PV值高-高报警	字/二进制	否*
16	Addr+17	PV值偏离报警(黄)	字/二进制	否*
17	Addr+20	PV值偏离报警(红)	字/二进制	否*
18	Addr+21	PV值变化率报警	字/二进制	否*
19	Addr+22	PV值报警滞后设定	字/二进制	否*
20	Addr+23	PV值误差死区设定	字/二进制	是
21	Addr+24	PV值模拟滤波常数设定	字/BCD	—
22	Addr+25	回路微分增益限幅设定	字/BCD	否**
23	Addr+26	SP值下限设定	字/二进制	是
24	Addr+27	SP值上限设定	字/二进制	是
25	Addr+30	控制输出值下限值设定	字/二进制	否**
26	Addr+31	控制输出值上限值设定	字/二进制	否**
27	Addr+32	远端SP值寄存器地址指针	字/十六进制	是
28	Addr+33	上升/保持设定标志	位	是
29	Addr+34	上升/保持编程表起始地址	字/十六进制	否**
30	Addr+35	上升/保持编程表错误标志	位	否**
31	Addr+36	PV值自动传送	字/十六进制	
32	Addr+37	预留	字/十六进制	

*仅当报警允许位由0→1时，才读数据。

**仅当PLC模式改变时，才读数据。

五、PID 控制模式设定 1 的位说明（Addr+00）

下表列出了PID模式设定1 (Addr+00) 各个位的定义情况。该字的用法可从本章后面的段落中得到。

Bit	PID控制模式设定1的说明	读/写	Bit=0	Bit=1
0	转换为手动模式请求	写	—	0→1, 发出请求
1	转换为自动模式请求	写	—	0→1, 发出请求
2	转换为串级模式请求	写	—	0→1, 发出请求
3	无扰动切换选择	写	模式 I	模式 II
4	控制方向选择	写	正向控制	反向控制
5	位置/速度算法选择	写	位置算法	速度算法
6	PV线性/平方根选择	写	线性	平方根
7	误差项线性/平方选择	写	线性	平方
8	误差死区控制选择	写	禁止	允许
9	微分增益限幅选择	写	Off	On
10	偏差冻结选择	写	Off	On
11	上升/保持操作选择	写	Off	On
12	PV报警监控选择	写	Off	On
13	PV偏离报警监控选择	写	Off	On
14	PV变化率报警监控选择	写	Off	On
15	设置为1时, 控制操作模式与CPU模式无关	不支持	—	—

※由于不支持模拟量自动传输功能, 故应将 Bit15 设置为 “0” (Off)。

六、PID 控制模式设定 2 的位说明（Addr+01）

下表列出了PID模式设定2(Addr+01)各个位的定义情况。该字的用法可从本章后面的段落中得到。

Bit	PID控制模式设定2的说明	读/写	Bit=0	Bit=1
0	输入与控制输出单极性/双极性选择 (见注1和2)	写	单极性	正负双极性
1	输入/输出数据(模拟量)格式选择 (见注1和2)	写	12位	15位
2	模拟量输入滤波功能	不支持	—	—
3	启用SP输入限幅	写	禁止	允许
4	重设时间(积分时间)单位选择	写	秒	分
5	自整定PID算法	写	闭环	开环
6	选择自整定	写	PID	仅PI(微分时间=0)
7	自整定开始	读/写	完成	开始
8	PID扫描时钟(内部用)	读	—	—
9	输入/输出数据格式16位选择 (见注1、2)	写	非16位	选择16位
10	输入输出分别选择独立的数据格式 (见注2、3)	写	相同格式	不同格式
11	控制输出范围单/双极性选择 (见注2、3)	写	单极性	正负双极性
12	输出数据格式选择(见注2、3)	写	12位	15位
13	输出数据格式16位选择 (见注2、3)	写	非16位	选择16位
14~15	预留	—	—	—

注1: 如果Bit9=0, 则以Bit0和Bit1设置的格式(正/正负、12/15位)读取数据; 如果Bit9=1, 则Bit0和Bit1的设置无效, 以Bit9定义数据格式(正范围、16位)。

注2: 如果Bit10=0, 则以Bit0、Bit1和Bit9的设置定义输入和输出范围以及数据格式(不读Bit11、Bit12和Bit13的值); 如果Bit10=1, 则Bit0、Bit1和Bit9只定义输入范围和数据格式, Bit11、Bit12和Bit13定义输出范围和数据格式。

注3: 如果Bit10=1并且Bit13=0, 则输出范围和数据格式由Bit11和Bit12(正/正负、12/15)的值定义; 如果Bit10=1并且Bit13=1, 则Bit11和Bit12的设置无效, 由Bit13定义数据格式(正范围、16位)。

七、模式/报警监控字（Addr+06）

下表列出了模式/报警监控字(Addr+06)各个位的定义情况。该字的用法可从本章后面的段落中得到。

Bit	模式/报警位说明	读/写	Bit=0	Bit=1
0	手动模式指示	读	—	手动
1	自动模式指示	读	—	自动
2	串级模式指示	读	—	串级
3	PV输入低-低报警	读	Off	On
4	PV输入低报警	读	Off	On
5	PV输入高报警	读	Off	On
6	PV输入高-高报警	读	Off	On
7	PV输入黄色偏差报警	读	Off	On
8	PV输入红色偏差报警	读	Off	On
9	PV输入变化率报警	读	Off	On
10	报警值设置错误	读	—	错误
11	PID控制运算值上溢/下溢	读	—	错误
12	自整定中	读	Off	On
13	自整定错误	读	—	错误
14-15	预留	—	—	—

八、上升/保持表标志（Addr+33）

下表列出了上升/保持(Addr+33)各个位的定义情况。该字的用法可从本章后面的段落中得到。

Bit	上升/保持标志位说明	读/写	Bit=0	Bit=1
0	启动上升/保持控制	写	—	0→1: 开始
1	锁定上升/保持步	写	—	0→1: 锁定
2	继续上升/保持步	写	—	0→1: 继续
3	强制上升/保持向下一步移动	写	—	0→1: 点动
4	上升/保持控制完成	读	—	完成
5	上升/保持PV错误(PV>SP)	读	Off	On
6	上升/保持当前步保持	读	Off	On
7	预留	读	—	—
8-15	使用8-15位表示当前步	读	按字节译码(十六进制)	

8~15位作为一个字节，存储当前上升/保持的步。该字节的值可以是1、2、3、4、5、6、7、8、9、A、B、C、D、E、F和10，分别代表了段1-16。如果此字节=0，那么上升/保持表不起作用。

九、上升/保持表地址（Addr+34）

在每个回路中，都有使用内置上升/保持数据配置目标值(设定点)的方法。要使用上升/保持功能，有必要在32个字的区域内设置合适的值。使用KPP软件对话框，可以很容易做到这一点。组态的每个回路都有专用于该回路的内藏的上升/保持程序控制。此功能产生了连续的SP值来进行控制。

在控制回路参数表中，Addr+34处指定上升/保持表的起始地址。该地址可以在用户存储器中的任何地方。如左图所示，不需要与控制回路参数表相邻。不管设置的步数为多少，每个上升/保持表都需要32个字。

上升/保持表参数按下表定义。详细说明见此章中上升/保持章节。

寄存器区		地址 偏移量	步	说明	地址 偏移量	步	说明
用户数据		+00	1	上升段结束SP值	+20	9	上升段结束SP值
R2000	回路 No.1 32 字	+01	1	上升段斜坡	+21	9	上升段斜坡
R2007	回路 No.2 32 字	+02	2	保持段持续时间	+22	10	保持段持续时间
		+03	2	保持段PV偏离	+23	10	保持段PV偏离
R3000	上升/保持 No.1 32 字	+04	3	上升段结束SP值	+24	11	上升段结束SP值
		+05	3	上升段斜坡	+25	11	上升段斜坡
		+06	4	保持段持续时间	+26	12	保持段持续时间
		+07	4	保持段PV偏离	+27	12	保持段PV偏离
		+10	5	上升段结束SP值	+30	13	上升段结束SP值
		+11	5	上升段斜坡	+31	13	上升段斜坡
		+12	6	保持段持续时间	+32	14	保持段持续时间
		+13	6	保持段PV偏离	+33	14	保持段PV偏离
		+14	7	上升段结束SP值	+34	15	上升段结束SP值
		+15	7	上升段斜坡	+35	15	上升段斜坡
		+16	8	保持段持续时间	+36	16	保持段持续时间
		+17	8	保持段PV偏离	+37	16	保持段PV偏离

R2034=3000(八进制)
指向上升/保持表的指针



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

十、上升/保持表设置错误标志（Addr+35）

下表列出了上升/保持设置错误标志字（Addr+35）的每个位的定义。在PID回路方式段落及这一章后面的PV报警章节中将给出详细说明。

Bit	上升/保持错误标志位说明	读/写	Bit=0	Bit=1
0	起始地址在寄存器范围下限溢出	读	—	错误
1	起始地址在寄存器范围上限溢出	读	—	错误
2-3	留作将来使用	—	—	—
4	起始地址在寄存器系统参数范围内	读	—	错误
5-15	留作将来使用	—	—	—

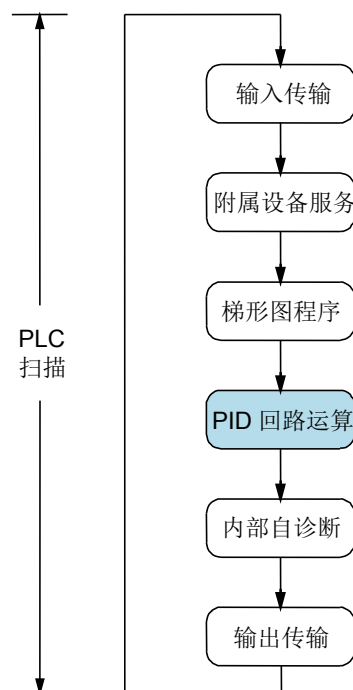
第三节 控制采样周期和时间表

一、控制采样周期 Addr+07

CPU的主要任务如右图所示的流程图。当CPU处于RUN模式下右图列出每次PLC扫描时所完成的任务。在梯形图程序之后进行PID回路运算。从用户的角度来看，梯形图程序运行时回路一直在进行运算。

一个控制回路的采样周期是PID运算周期。每次运算都产生一个新的控制输出值。可以把一个回路的采样周期设定为50ms~99.99s。因此大多数回路不需要在每次PLC扫描都进行PID运算。一些回路可能在每1000次扫描后才需一次PID运算。

可以选择每个回路合适的采样周期，CPU将自动安排并在合适的扫描周期执行PID运算。



注意：由于不支持模拟量输入/输出的自动传送，因此需要使用梯形图程序进行写入。
通过在Addr+00的Bit15写入1，来实现在梯形图程序中直接访问控制回路参数表中Addr+36、Addr+37内容。

二、选择最佳的采样周期

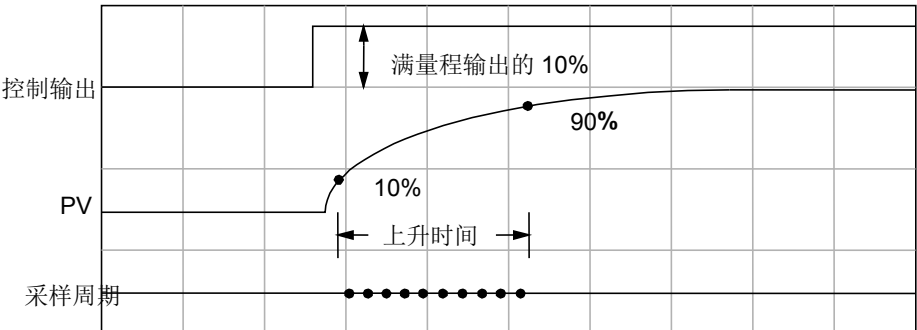
不存在适合任意控制回路的最佳采样周期。一个好的采样周期是为了能同时满足各种要求的一个折衷值：

- 为控制输出变化，采样周期要与PV值的响应时间成正比。通常，对长期过程使用较慢的采样周期，对于短时过程使用较快的采样周期。
- 较快的采样周期会提供比较平滑的控制输出和精确的PV特性，但需要更多的CPU处理时间。
- 较慢的采样周期提供了粗略的控制输出和不太精确的PV特性，但使用的CPU处理时间更少一些。
- 采样周期过慢将导致系统不稳定，尤其是当设定值发生改变或发生干扰时。

首先，为回路确定的采样周期要足够快，以避免控制不稳定（这点尤为重要）。按下页的步骤找出一个初始采样周期。

三、确定合适的采样周期（Addr+07）

- 1. 在过程开环的情况下，（不对PID控制进行设置）将CPU转换为RUN模式。（如果已经对PID控制进行了设置，将PID控制转换到手动模式。）手动设置控制输出值，使PV稳定在安全的范围内。
- 2. 当过程的外部干扰可忽略的时，选择一个时间，然后对控制值施加一个10%的阶跃变化。
- 3. 记录PV的上升或下降时间（10%~90%两点之间的时间）。
- 4. 把记下的上升或下降时间除以10，就可以作为开始调整PID控制回路的初始采样周期。



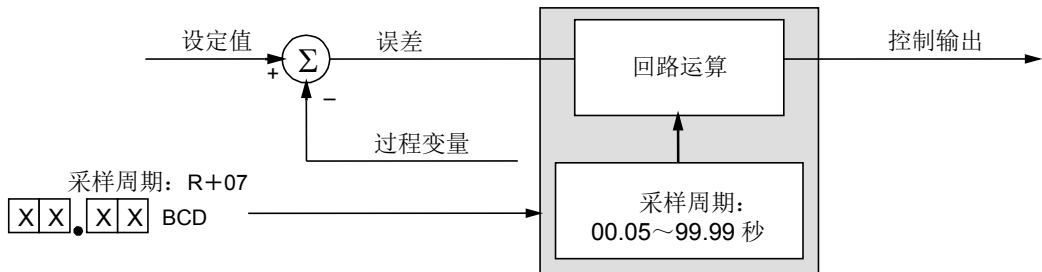
上图中，假设测量到的PV上升时间响应是25s。根据此测量值，则采样周期为2.5s。按上图所示，在整个上升时间内，采样周期对应的轴显示了十个采样点。这表示了PV值变化时PID运算的次数。当然，在运行过程中，PID运算会根据采样周期连续进行。



注意：采样周期过快将使PV变化率报警并降低此报警的分辨能力，因为报警值是根据每个采样周期中PV的变化而确定的。例如，PV变化率最小采样周期为50ms，即每秒20个PV值，每分钟1200个PV值。

四、采样周期编程

每个回路的控制回路参数表都有一个采样周期的数据地址。参见下图，地址R+07包含一个00.05~99.99（带一个隐含的小数点）的BCD码，表示50ms~99.99s。该数值可以通过KPP的PID 设置画面进行设置，也可以用直接写入寄存器方法来设置。在PID控制正确动作之前，必须进行设置。



五、CPU 扫描对 PID 回路的影响

因为PID控制是CPU扫描任务的一部分，所以使用PID控制将增加CPU扫描时间。扫描时间的增加量与使用的PID控制回路数和每个PID控制回路的采样周期成正比。

另外，单个PID控制回路的执行时间与选择项目数量有关，例如报警、平方根误差等，右表给出了预想的时间范围。

PID运算时间

最小	150 μS
标准	250 μS
最大	350 μS

为了计算扫描时间的增加量，有必要知道仅梯形图程序（无PID回路）的扫描时间。当使用相同的PID回路运算时，最小扫描时间增加的百分比将比最大扫描时间更多。平均扫描时间的计算公式如下：

$$\text{带 PID 回路的平均扫描时间} = \left[\frac{\text{无 PID 回路的扫描时间}}{\text{回路采样周期}} \times \text{PID 运算时间} \right] + \text{无 PID 回路的扫描时间}$$

根据上面的公式，增加慢速采样周期的回路时，对扫描时间的影响很小。接下来，根据上面的公式，讨论增加各种PID控制回路数对CPU扫描时间的影响。

$$\text{带 PID 回路的平均扫描时间} = \left[\sum_{n=1}^{n=L} \frac{\text{无 PID 回路的扫描时间}}{\text{第 } n \text{ 个回路采样周期}} \times \text{PID 运算时间} \right] + \text{无 PID 回路的扫描时间}$$

在上面的新公式中，计算了从1到L（最后一个控制回路）PID控制所增加的总时间，最后再加上最右边一项“无PID回路时的扫描时间”。假定有一台SJ-Ether PLC控制四个回路，无PID回路的扫描时间为50ms。下表显示了每个PID控制回路的数据和求和项值。

PID控制回路号	说明	采样周期	求和项
1	进气阀蒸汽流量	0.25秒	50 μs
2	水槽温度	30秒	0.42 μs
3	主罐染料液位	10秒	1.25 μs
4	高压釜蒸汽压力	1.5秒	8.3 μs

现在把求和项加起来，再加上无PID回路的扫描时间值，平均扫描时间为：

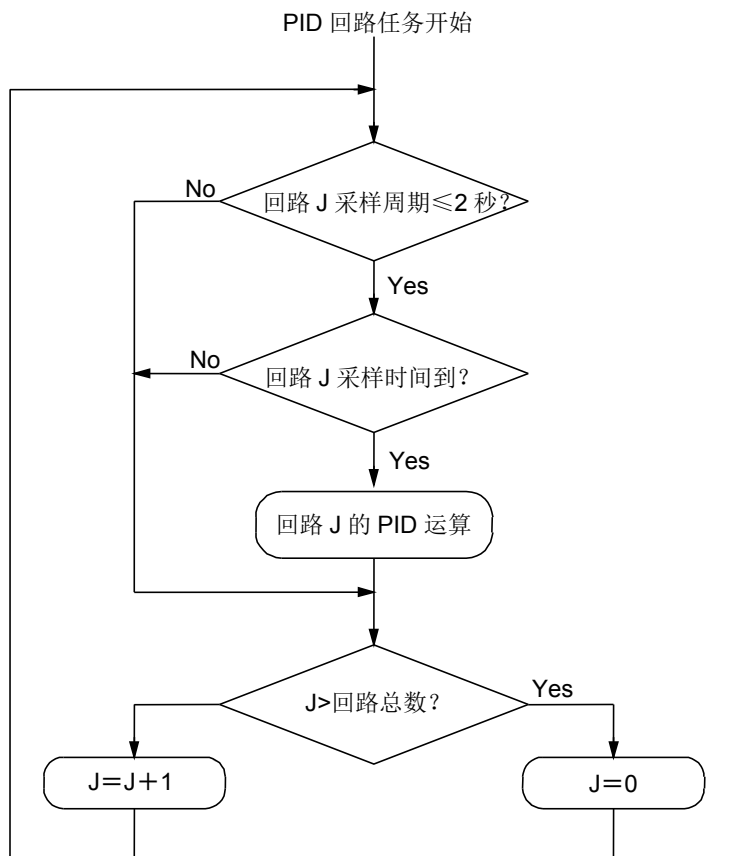
$$\text{带 PID 回路的平均扫描时间} = \left[50 \mu s + 0.42 \mu s + 1.25 \mu s + 8.3 \mu s \right] + 50 \text{ms} = 50.06 \text{ms}$$

CPU仅在采样的扫描时才进行PID运算。内置的PID控制调度程序的应用遵循下列规则：

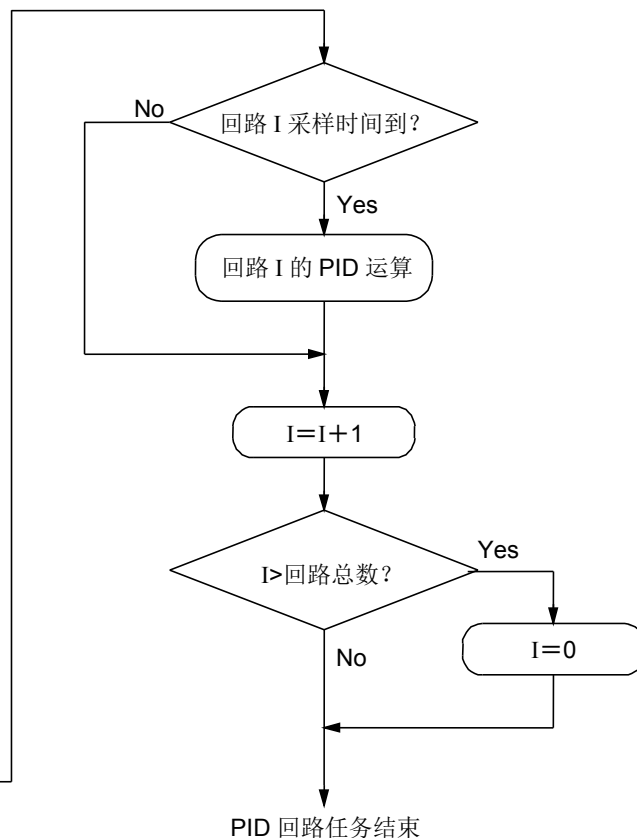
- 采样周期≤2s时，为保持各回路的采样周期，将使用更高的采样率进行PID控制。采用高采样率进行PID控制，会增加PLC的扫描时间。因此，仅在需要时采用高采样率。
- 采样周期>2s时，为保持各回路的采样周期，最低采样率为每次扫描一个或几个PID回路。

PID控制的调度如下面的流程图所示，这是对CPU扫描流程图中“PID回路运算”任务的更详细说明。指针“I”和“J”分别对应于慢速PID控制（>2s）和快速PID控制（≤2s）。如果指定了一些快速PID控制，流程图中指针J从回路1到最后一个回路递增。指针I每次扫描时只递增一次，控制慢速PID回路刷新。除了采用不同的采样周期，指针I和J都从1到最大回路数循环往复，它们的组合保持所有PID回路的彻底刷新。

回路采样时间≤2秒：



回路采样时间>2秒：



第四节 过程控制成功的十个步骤

CPU具有卓越的过程控制功能。自控系统的调试有一定的难度，因为某个故障现象可能是由许多原因引起的。在构建新的PID控制回路时，我们建议按步骤实施下面的方法：

步骤1：了解配方

最重要的知识是如何生产产品，这些知识是设计一个有效的控制系统的基础。对于建立一个好的过程的“配方”，以下内容很重要：

- 确定所有相关的过程变量，如温度、压力、或流量等需要精确控制的量。
- 给出一个过程周期内每个过程变量的设定值。

步骤2：设计控制计划

这意味着设备控制过程变量跟随设定值，涉及到许多因素，如能效、设备成本、生产中调整设备的能力等，还必须确定在过程中如何产生设定值，以及设备操作员是否可以改变SP值。

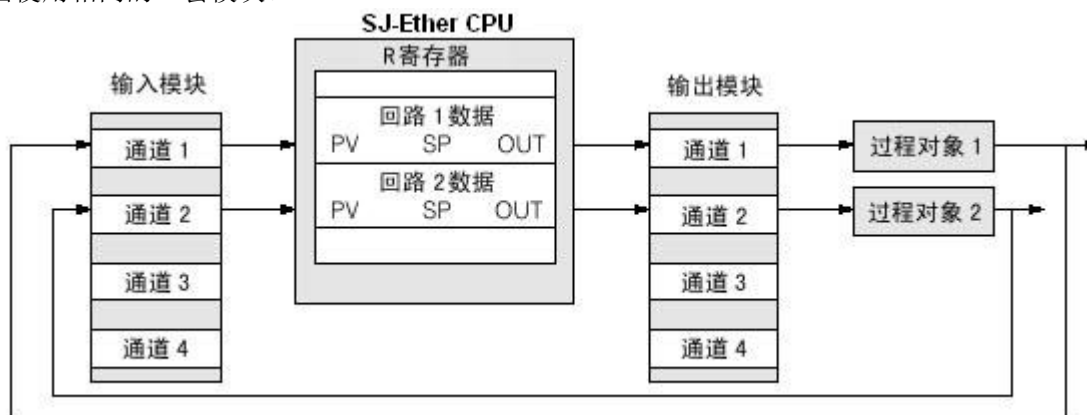
步骤3：确定元件的规格和测量范围

假定控制计划有效，请选择合适的执行机构和相应的传感器。

- 请选择与负载大小相匹配的执行机构（加热器、泵等）。大于负载的执行机构，在SP变化后将过程产生重大影响。规格过小的执行机构，会导致PV滞后，在SP变化后，达不到设定或导致过程不稳定。
- 选择与过程的相匹配的PV传感器。确定PV所需的控制精度（如2℃以内），确保传感器的输入值的分辨率（最低有效位）至少是控制精度的5倍。但是，传感器过分灵敏可能会导致控制上的波动等。SJ-Ether提供了12位和15位，单极性和正负双极性数据格式选择以及16位单极性选择。这些数据格式用于SP、PV、控制输出和积分项。

步骤4：选择I/O模块

在确定了PID控制回路数、测量值PV和SP值后，就可以选择合适的I/O模块。参见下面的图，在多数情况下，几个控制回路可以使用相同的输入/输出模块。示例中两个回路的PV值和SP值和控制输出使用相同的一套模块。



步骤5：接线和安装

在选择并购置了所有控制部件和输入/输出模块后，就可以进行接线和安装。安装PID控制回路时，最常见的接线错误是：

- 颠倒了传感器和执行机构接线端的极性。
- 控制部品之间的信号接地不适当。

步骤6：控制参数

接线安装完成后，设置控制参数。设置控制回路参数表的最简单的办法是使用KPP软件。该软件提供了PID设置对话框，从而简化了这项工作。



注意：在选择键入的数值之前，先理解本章中提到的所有回路参数的含义是十分重要的。

步骤7：检查开环执行情况

一旦传感器和执行机构接线完成，输入PID控制参数后，必须手动(使用手动模式)仔细检查新的控制系统。

- 确定来自传感器的PV值是否正确。
- 如果一切正常，逐渐增加控制输出，看PV是否正确响应(并按正确的方向移动)。

步骤8：PID控制回路调整

如果开环调试中，PV读数正确并且控制输出正确作用于过程，则可进行闭环调整。(在这一步，通过PID控制调整，PV自动跟随SP。)

步骤9：运行PID控制过程循环

如果闭环调试中，PV能跟踪SP的细微变化，则可以在实际环境运行PID控制过程，用编程来实时产生SP值。在这一步中，根据配方调整SP，使用产品对设备进行测试。

步骤10：保存参数

完成控制回路调试和调整工作后，一定要把所有PID控制参数保存到磁盘。



警告：要确保在过程失控时，紧急停止和断电装置能迅速动作。某些过程的失控，可能会导致设备损坏或严重的人员伤害。

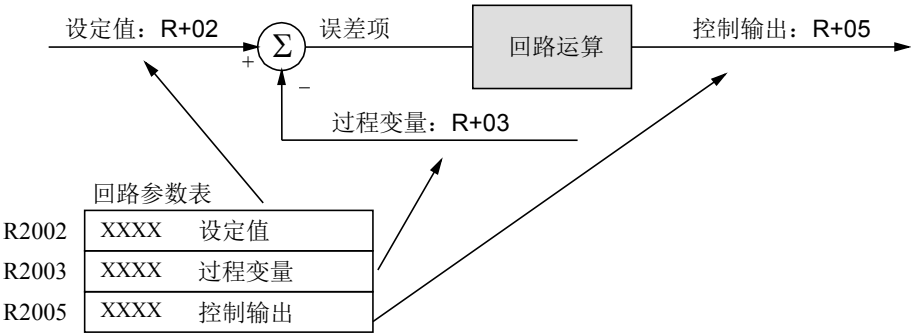


注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

第五节 基本回路动作

一、数据地址

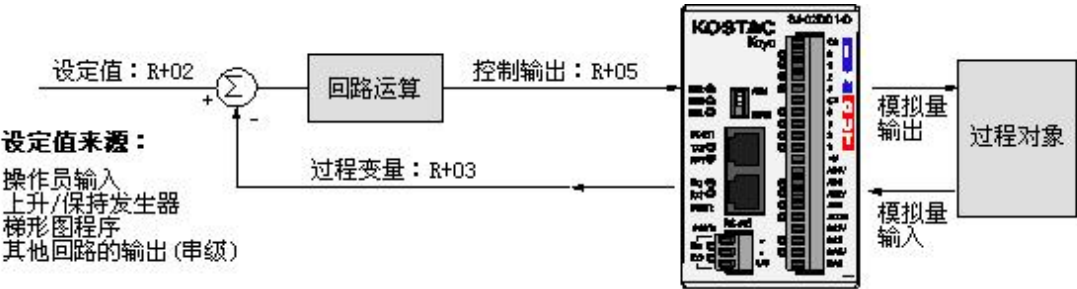
每个PID控制回路都取决于它对应的控制回路参数表的使用情况和数据值。下图显示了控制回路参数表地址中三个主要回路变量：SP、PV和控制输出的地址。假设回路参数表起始地址为R2000(用户可以选择其它地址), SP、PV和控制输出分配的地址如下所示。



二、数据传送

SP、PV和控制输出的相关数据必须连接到设备。在下图中，显示了PID控制变量的来源和目的地。通过模拟量输入模块，将过程信息读入PID控制参数表的过程变量；通过模拟量输出模块，将控制输出的值作用于过程。

梯形图程序需要通过模拟量模块将多个控制回路的数据读入PID控制参数表，也需要反过来通过模拟量模块将CPU的数据输出。



如上所示，设定值有几个来源。根据PID控制模式，许多应用程序在不同的时间，使用两种或两种以上的设定值来源。另外，回路控制操作和编程方法决定了如何产生设定值。

当使用内置的上升/保持发生器或串联回路时，PID控制器自动将设定值写入R+02中。

如果打算使用其他源来写入设定值，必须通过梯形图程序将设定值写入控制参数表的R+02。

显然，三个主要回路参数在特定的时间里各自只能有一个来源或去向。在开发应用时，最好先画出回路控制框图来表示数据来源，以避免发生错误。

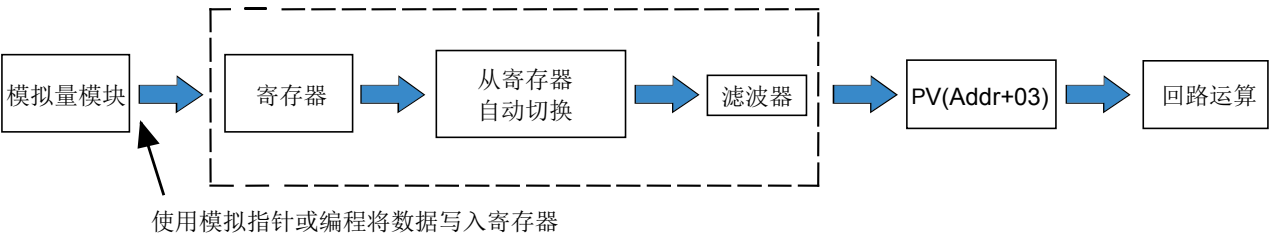
三、使用滤波器选项的 PV 自动传输功能

内置滤波器采用以下算法。

$$y_i = k(x_i - y_{i-1}) + y_{i-1}$$

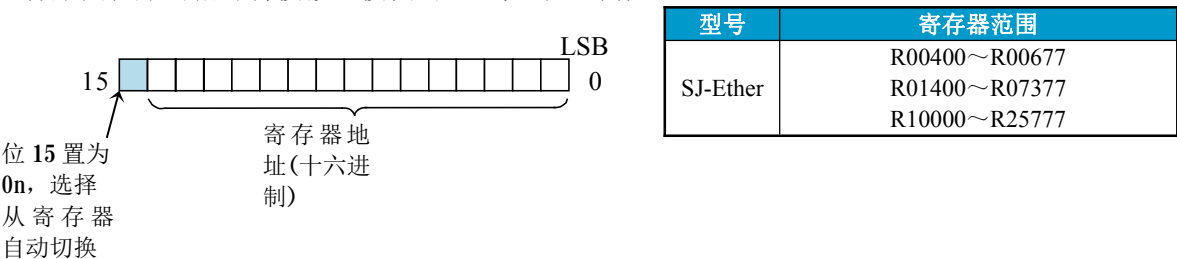
- y_i = 滤波器的当前输出
- x_i = 滤波器的当前输入

- y_{i-1} =滤波器前一次输出
- k =PV模拟量输入滤波器系数
- 使用梯形图程序读取使用滤波器的寄存器，写入指定的寄存器。



来自寄存器的PV自动传送(Addr+36)

下表列出了寄存器传输选项中PV自动传输字(Addr+36)的直接定义。使用此选项，可以使用梯形图程序的指针将模拟量模块的通道值写入寄存器。

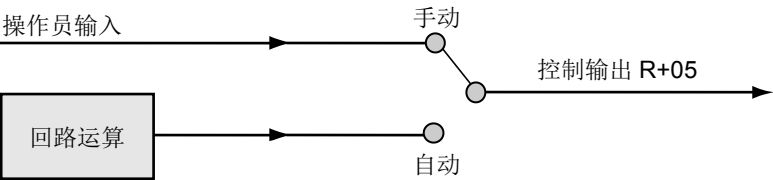


有三种标准控制模式：手动、自动和串级。在不同模式下，三个基本变量SP、PV和控制输出来源是不同的。下面介绍三种控制方式和其信号源。

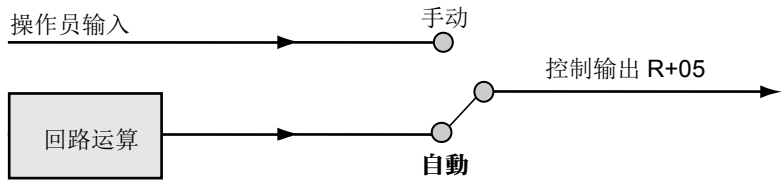
如下图所示，PID控制可以从一种模式更改为另一种模式，但是不能从手动模式直接转到串级模式，反之亦然。从手动模式直接转到串级模式，或反之，PID控制需要同时更改两个数据源，因此禁止该模式更改，可能会导致失控，故禁止这两种模式转换。



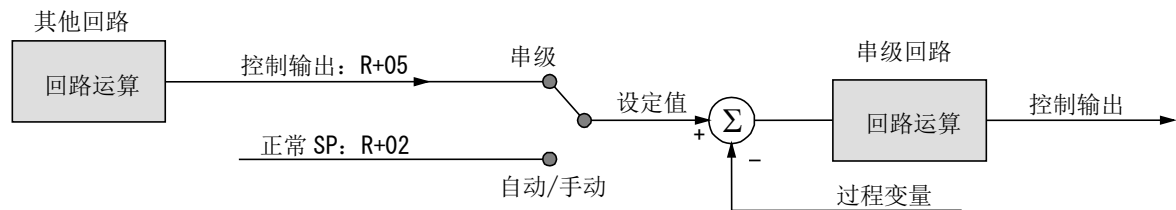
在手动模式下，回路不进行PID控制(但是，回路报警仍起作用)，CPU不会将数值写到控制回路参数表的R+05中。为了保持过程控制，操作员等监控PV值，并手动将数据写入Addr+05中。下图显示了等效于手动模式操作的示意图。



在自动模式下,PID控制回路正常运行并产生新的控制输出值。回路每个采样周期都进行PID运算并将结果写到R+05中,如下图所示。

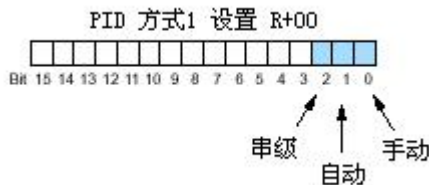


在串级模式下,与自动模式下的动作相同,但有一个重要区别。对于SP源数据,使用另一个PID回路的控制输出值,而不使用控制参数表R+02的值。因此,在自动或手动模式下,PID控制使用R+02的数据,但在串级模式下,PID控制读取另一个控制参数表的控制输出。



五、控制模式变更方法

PID控制设置1(R+00)的最低三位对应回路运行模式的请求位。



这些控制模式请求位在正常情况下是“000”。要改变控制模式,把相应位设成“1”必须保持一个扫描周期。PID回路控制器收到模式改变请求后,会自动把这些位复位成“000”。



注意: 这些位不是命令,而是模式改变请求(有些条件下,会禁止某些模式改变,请参阅下一页)。

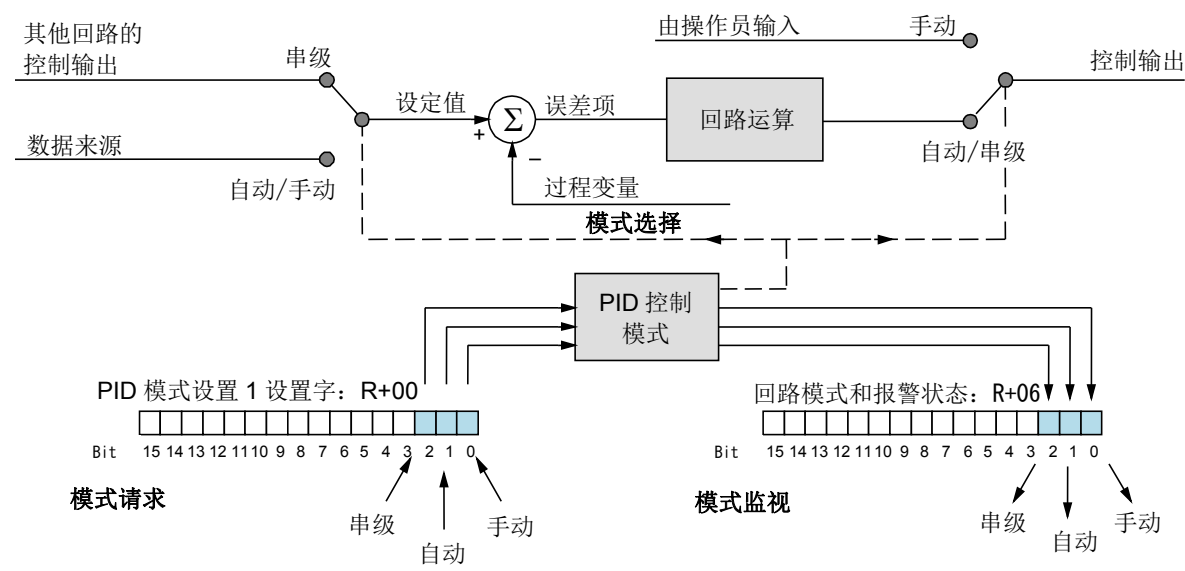
控制模式改变的请求方法如下:

- 使用KPP的PID监视视图是最简单的方法。点击其中一个单选按钮,设置相应的位。
- 使用KPP,监视R+00的内容,其数据是一个4位BCD/十六进制值。将该值与模式位进行或运算后写入R+00。
- 当PLC处于运行模式时,梯形图程序可以请求任何回路控制模式。这在应用程序启动后是十分必要的。如右图所示,在程序中使用SET指令(不使用OUT指令)设置模式位。当I0由0变为1时,回路将自动位设置成1,回路控制器会将其复位。
- 作为一般方法,将梯形图程序与操作面板连接起来,然后设置模式位。



注意: DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

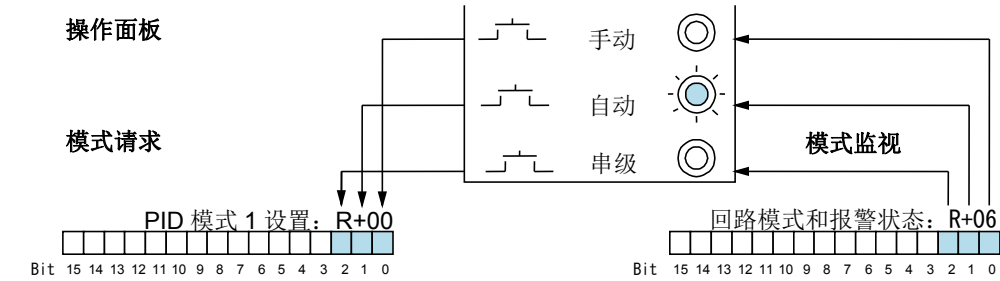
我们只能请求控制模式的改变，PID回路控制器决定是允许模式更改，还是通知控制模式的状态。控制回路参数表的R+06为回路模式与报警状态字，其位0、1、2表示当前控制模式。请求/监视功能如下图所示。该图显示了两中可能的模式：一般SP数据源和SP数据来自其他回路的控制输出。



六、使用操作面板控制 PID 模式

由于手动、自动和串级模式是最基本和重要的PID控制，因此操作面板上可能需要“硬接线”的模式控制开关。大多数应用程序只需要手动和自动两种选择(串级用于某些高级应用程序)。要记住，模式控制是模式更改请求位，实际控制模式存储在其他地址。

下图显示了使用操作面板按钮请求更改PID模式的例子。操作面板的模式显示没有连接到开关，而是连接到相应的数据位置。



七、PLC 模式对控制模式的影响

当选择了PID控制与PLC模式关联的选项，PLC模式(停止、运行)将会与PID控制相互影响。两者的相互影响，概括如下：

- PLC处于停止模式时，所有PID控制回路都处于手动模式且不进行回路运算。输出模块（包括模拟量输出）在此方式下没有输出。所以此方式下，手动控制是不可能的。
- 只有PLC在运行模式下，CPU才允许更改回路控制模式。CPU把16条回路的控制模式记录下来。如果PLC运行模式下突然掉电，恢复供电后，则CPU将恢复所有回路先前的模式（手动、自动或串级）。
- 当从停止切换到运行模式时，CPU将强制每条回路返回到之前PLC运行模式下的控制方

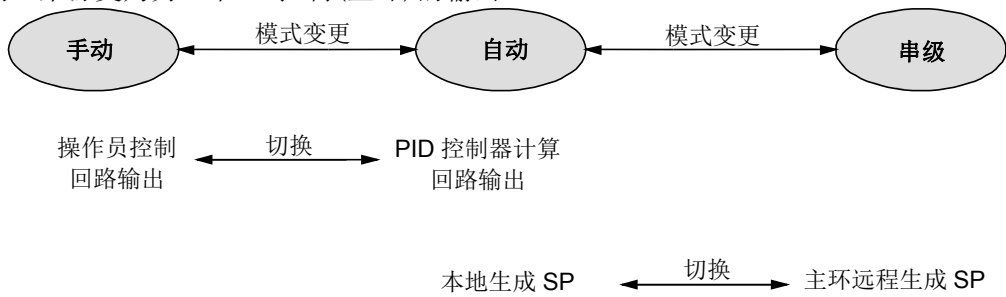
- 式。
- 只有当CPU在停止模式时，才能添加和设置新的回路，新回路自动以手动模式开始运行。

八、回路控制模式的覆盖

- 一般情况下，通过R+00的请求位0、1和2，确定回路的控制模式。但在某些情况下，即使请求模式变更，也无法改变模式：
- PID控制模式与PLC模式相关联，而PLC处于STOP模式。
 - 当主环处于串级模式时，组成串级的副环不能从手动模式转换为自动模式。在其他情况下，PID控制会自动改变控制模式以保证安全运行。
 - PID控制发生错误后，自动转换到手动模式。
 - 如果在串级模式下，PID控制组成串级的副环发生故障，则主环自动变为手动模式。

九、无扰动切换

在过程控制中，“切换”一词具有特定的含义。当改变PID控制的操作模式时，如下图所示进行切换。当改变PID控制模式时，要设置一些控制参数。当控制回路从手动模式切换为自动模式时，由控制器对控制输出进行控制。当控制回路从自动模式切换为串级模式时，自动模式下的SP来源变为另一个PID控制(主环)的输出。



控制转移的基本问题是不同的控制模式使用不同的数据源。这会出现PID算法不希望的阶跃变化，使控制输出中出现“冲击”，导致PID控制失控。当通过“无扰动切换”功能改变控制模式时，强制使一个参数等于另一个参数，切换变得平滑(控制输出中没有突变)。

SJ-Ether PID控制器的无扰动切换有两种类型：无扰动切换 I 和无扰动切换 II。可以使用KPP软件的PID设置对话框选择转换类型，也可以使用图示的PID模式1 R+00的Bit3设置。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

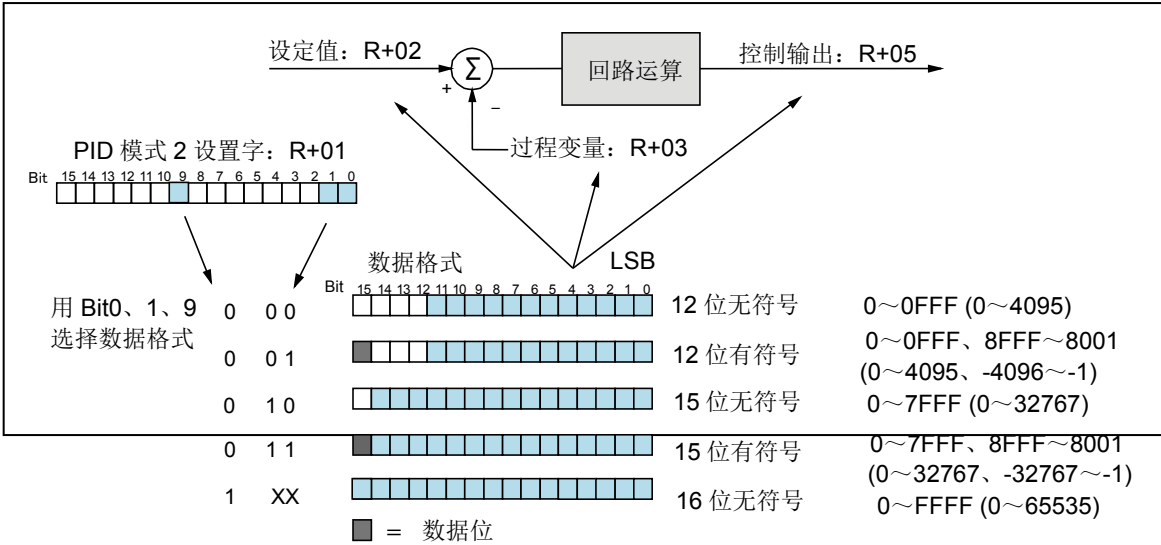
无扰动切换 I 和 II 的转换类型的特点如下表所示。注意，PID方程使用的算法是位置型还是速度型。如果使用PID速度型算法时，必须使用无扰动切换方式I。

转换类型	切换选择位	PID算法	手动→自动模式切换	自动→串级模式切换
无扰动切换 I	0	位置型	使偏差项=控制输出 使SP=PV	使主环输出=副环PV
		速度型	使SP=PV	使主环输出=副环PV
无扰动切换 II	1	位置型	使偏差项=控制输出	无
		速度型	无	无

第六节 PID控制的数据构成

一、控制参数的数据格式

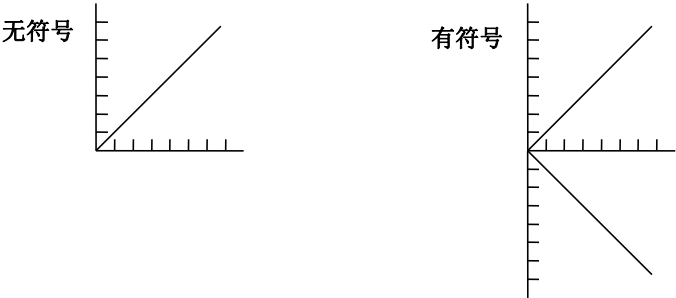
PID控制主要有三种控制变量 (SP、PV和控制输出)，其数据格式由过程变量范围和分辨率的选择决定 (R+04的偏差值也使用此数据格式)。数据格式可选择位长度为12、15、16位，还可选择无符号或有符号 (最高位表示符号)。通过对PID模式2 (R+01) 的0、1、9位的选择，可以有五种组合。当从菜单中选择数据格式时，KPP的PID设置对话框会自动设置这些位。



可以选择各种数据格式。数据格式将确定PID控制回路、PV传感器和控制输出设备之间的数字接口。此外，设定值也会采用相同的数据格式。通常情况下，在最初构成PID控制时选择数据格式，此后不再变更。

二、选择有符号或无符号

在选择数据格式时，请选择是有符号还是无符号。对于大多数应用，如温度控制，只使用正数，因此选择无符号。通常，是否选择有符号，取决于控制输出。例如，速度型算法可能包含正向、反向控制。速度设定值为0时，理想的控制输出也是0。在这种情况下，必须使用有符号格式。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

三、数据偏移

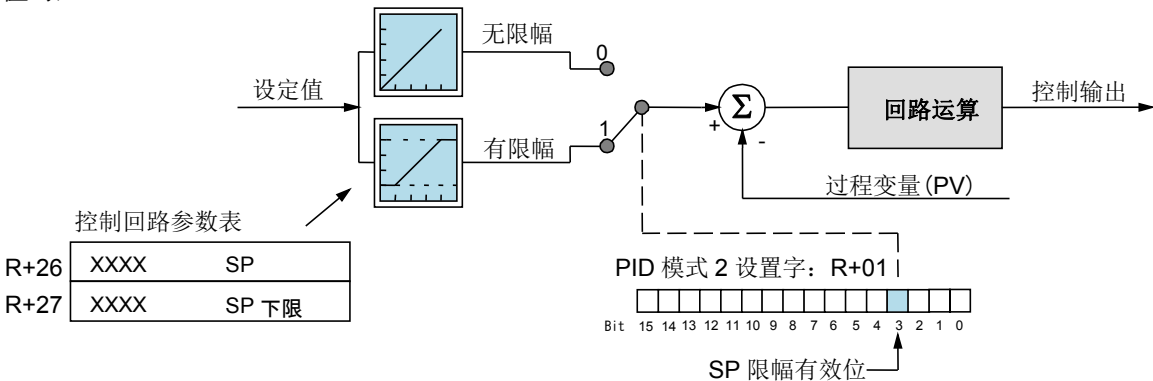
在许多批处理应用中，传感器和执行机构使用4-20mA信号连接到SJ-Ether模拟量模块。由于零点是4mA而不是0mA，因此该信号类型有20%偏移量。但是模拟量模块会将信号转换为数据，同时删除偏移量。例如，一个4-20mA信号被转换为十六进制数0000-0FFF，或十进制数0-4095。在这种情况下，请选择12位无符号数据格式，并使用梯形图程序在PID控制表和模拟量模块之间正确传送数据。

- PV的偏移量：如果PV值有了20%的偏移量，则减去其上限值的20%，再乘以1.25，使偏移量为零。
- 控制输出：控制输出送给设备的数据有20%偏移量，则从手动向自动模式切换之前，用梯形图程序把与偏移量相等的值写入偏置寄存器(R+04)。PID控制回路将此偏移量视为是过程的一部分，自动地监视它。

四、设定值（SP）的限幅

控制回路参数表位置R+02中的设定值是过程变量的目标值。选择了变量的数据格式后，可以设置PID控制中使用的SP值范围。由于许多回路在不同时间重写设定值，因此有两个或两个以上数据源。设置限幅值可保护过程免受不良SP值的影响。

下图中，通过将PID模式2(R+01)的位3置0n，使SP限幅功能有效。限幅功能有效后，控制回路参数表中的R+26和R+27分别存放SP下限和上限。PID控制在内部使用限幅值，可以随时将任意值写入R+02。

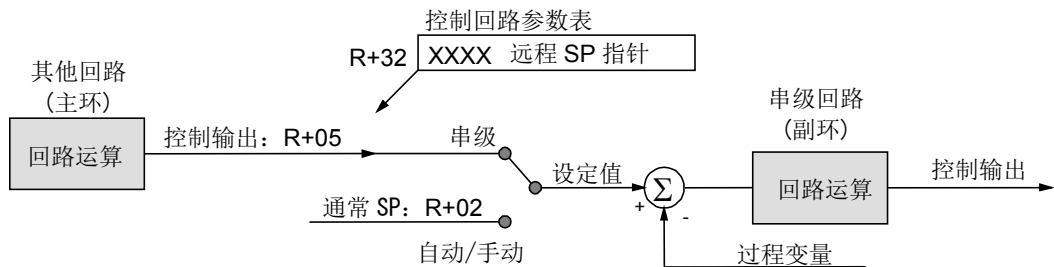


PID控制在每次计算前检查这些SP的上限和下限。这意味着控制过程中，可以通过梯形图程序改变限幅值，可以限制SP输入值。

五、远程设定值（SP）地址

PID控制器可以在当前控制模式下在两个数据源之间进行选择，如下图所示。当自动或手动模式时，回路从控制回路参数表的R+02读取设定值。如果使用级联模式进行控制，则必须梯形图程序将数据写入控制回路参数表中的远程SP指针。

远程SP指针位于控制回路参数表R+32。在串联控制时，必须使用梯形图程序，将包含主环的控制输出的地址写入R+32。找出主环参数表的起始地址并加上偏移量+05。

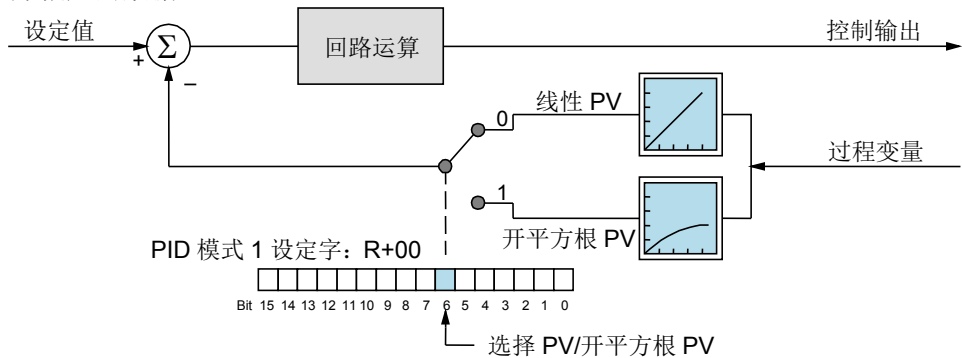


如果知道地址，可以使用KPP的PID控制对话框轻松将其设置为远程SP指针。

六、过程变量（PV）的设置

PID控制的目的是使输入到控制器的过程变量等于设定值(SP), 并尽快随设定值的改变而改变。测量过程变量的多数传感器具有线性响应曲线，大多数温度传感器在其传感范围内基本上是线性的。但是，使用孔板技术的流量检测给出的信号是流量的平方。因此，在线性控制系统(如PID)中使用此信号之前，需要取平方根的功能。

某些流量传感器有开平方功能，但这增加了传感器的成本。PID回路(PV)输入值可选用开平方功能，如下图所示。用PID模式设置字R+00的Bit6，可选择使用常规的(线性)PV数据，还是开平方根后的数据。



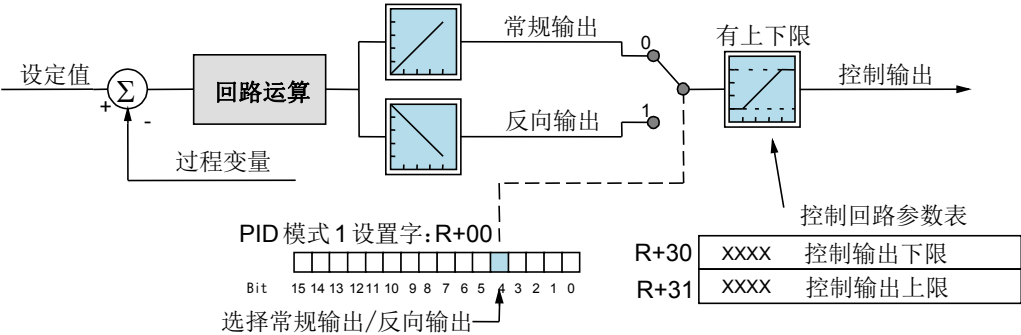
注意：当使用开平方根PV时，需要调整SP范围。过程变量的平方根作为PV输入，PID控制器控制输出。将模拟量值的平方根除以SP值，并将结果写入存放SP值的R+02。这样会降低SP的分辨率，但许多流量控制不需太精确。根据使用的数据格式，使用以下公式之一计算SP，在允许范围的开始处设置SP上限是个好主意。

数据格式	SP比例	SP范围	PV范围
12位	SP=PV输入/64	0-64	0-4095
15位	SP=PV输入/181.02	0-181	0-32767
16位	SP=PV输入/256.00	0-256	0-65535

七、控制输出设置

控制输出是PID运算的数字结果。所有其他选择的参数都会影响控制回路的控件输出值。以下可选项会影响最终的控制输出。图的最右侧是最终输出，可以通过设置下限值和上限值进行限制。使用KPP软件的PID设置对话框可以设置R+30和R+31的值。

当PID控制异常时，控制输出的下限值和上限值可以保证输出在有限范围。但是，请勿将上下限用于限制可能会损坏机器的移动(请改用限位开关等)。



其他选择包括常规/反向输出，使用PID模式1设置字R+00的Bit4设置输出方向。无论是无符号还是有符号，常规控制的输出上升时，在误差项产生负偏差(偏差=(SP - PV)),使输出下降。反向输出与常规输出相反。

常规/反向输出选择用于构成正向/反向控制回路。这种选择由控制输出方向、过程变量的响应方向等决定。有关正向与反向动作控制的更多信息，参见PID运算段落中正/反作用回路内容。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

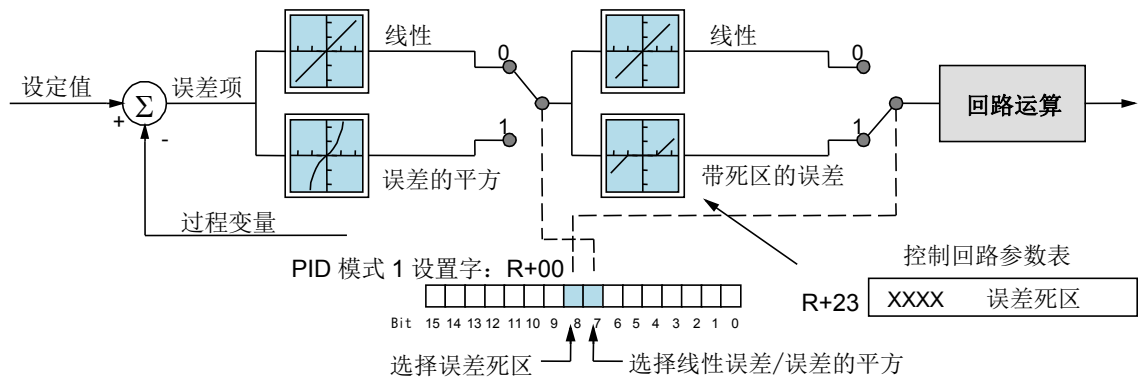
八、误差项的设置

误差项位于CPU的PID控制器内部，参与PID运算。此数据不能直接存取，但可以通过减法算出：误差=(SP-PV)。如果选择PV开平方根，误差=(SP-平方根(PV))。误差与正负号决定了每次PID运算的控制输出的下一个变化。

误差项有几种选项，参见下图。PID模式1设置字R+00的第7位选择线性或误差平方，第8位启用或禁用误差死区。



注意：当第一次设置时，最好是使用标准误差项。回路调整后，如果这些功能会增强控制，就可以使用。



误差平方：选择误差平方功能后，只简单地对误差项平方(但保留原来的代数符号)。通过减小对较小误差值的响应, 并保持对较大误差值的响应, 来影响控制输出。误差平方项用于下列一些情况：

- 含有干扰的PV信号：使用误差项的平方，可以减少低频噪声对PV值的影响。平方误差保持了对大误差值的响应。
- 非线性过程：某些过程(如PH值控制)需要非线性控制器以得到最佳效果。另一个应用是缓冲罐的控制，其控制输出信号必须是平滑的。

误差死区：选择误差死区功能后，如果误差值范围接近零，PID控制器使用零作为误差值进行运算。如果误差值大于死区范围，则用误差值减去死区后进行运算。

有必要将死区数值写入控制回路参数表R+23。SP和PV单位相同(在12位模式下0~FFF，15位模式下0~7FFF)，PID回路控制器自动地将死区以零误差点为中心，对称分布。

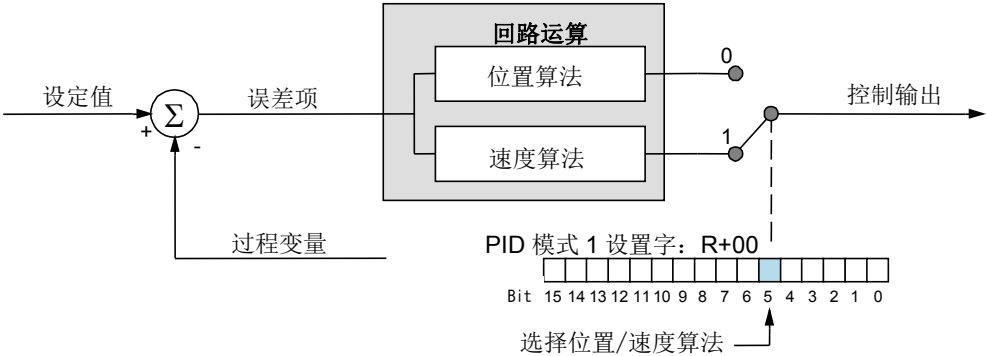
第七节 PID算法

PID算法广泛用于过程控制。无论是模拟量信号还是数字量信号，PID控制方法都能很好地提供电子解决方案。SJ-Ether CPU通过求解基本方程，进行PID控制。输入/输出模块可用于将电气信号转换为数字形式(反之亦然)。

SJ-Ether有两种PID控制类型：“位置类型”和“速度类型”，根据不同应用进行选择：

- PID位置算法：用SP减去PV得到的差(误差)，计算控制输出。
- PID速度算法：用PV达到SP的变化率(速度)，计算控制输出。

多数应用使用PID位置算法。如果无法确定使用哪种算法，就先试试位置算法。用KPP的PID视图设置对话框选择算法，也可以通过设置PID模式1设置字R+00的Bit5来选择算法。



注意：选择PID算法是控制回路工作的基础，一旦回路设置好之后，一般不再被改变。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

一、位置算法

$$M_n = K_c * e_n + K_i * \sum_{i=1}^n e_i + K_r * (e_n - e_{n-1}) + M_o$$

选择位置算法后，用以下公计算控制输出Mn：

在上面的公式中，积分项的总和与初始输出值相加成为“偏差”项(Mx)。用偏差项，定义偏差与控制输出的采样时间函数：

位置 关变量	$M_{X0} = M_o$ $M_{Xn} = K_i * e_n + M_{Xn-1}$	置算法变量及相 如下： Ts=采样 Kc=比例
周期	$M_n = K_i * \sum_{i=1}^n e_i + M_o$	
增益	$M_n = K_c * e_n + K_r * (e_n - e_{n-1}) + M_{Xn} \dots$	第“n”次采样时刻输出

$K_i = K_c * (T_s / T_i)$ 积分项系数

$K_r = K_c * (T_d / T_s)$ 微分项系数

T_i = 积分时间常数

T_d = 微分时间常数

SP_n = 第“n”次采样时刻的设定值 (SP值)

PV_n = 第“n”次采样时刻的过程变量 (PV)

$e_n = SP_n - PV_n$ = 第“n”次采样时刻的误差项

M_0 = 初始控制输出值

M_n = 第“n”次采样时刻的控制输出

这些公式可在有关控制过程的大多数文献资料中找到。如下图所示，PID位置算法被分成了几部分，分别是P、I、D项以及偏差项。



初始输出是假定当回路从手动方式切换到自动模式时的输出值。初始输出和积分条件的总和就是偏差条件，该项保持输出的“位置”。因此，下面描述的速度算法没有偏差项。

二、速度算法

PID公式的速度算法可以通过位置算法的第n次算式减去第(n-1)次算式的变换得到。

速度算法的变量和相关变量如下：

T_s = 采样周期

K_c = 比例增益

$K_i = K_c * (T_s / T_i)$ 积分项系数

$K_r = K_c * (T_d / T_s)$ 微分项系数

T_i = 积分时间常数

T_d = 微分时间常数

SP_n = 第“n”次采样时刻的设定值 (SP值)

PV_n = 第“n”次采样时刻的过程变量 (PV)

$e_n = SP_n - PV_n$ = 第“n”次采样时刻的误差项

M_n = 第“n”次采样时刻的控制输出

PID速度算法的公式如下：

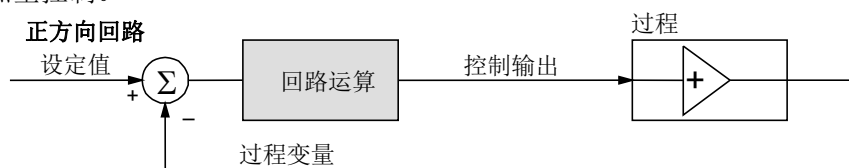
$$\Delta M_n = M_n - M_{n-1}$$

$$\Delta M_n = K_c * (e_n - e_{n-1}) + K_i * e_n + K_r * (e_n - 2 * e_{n-1} + e_{n-2})$$

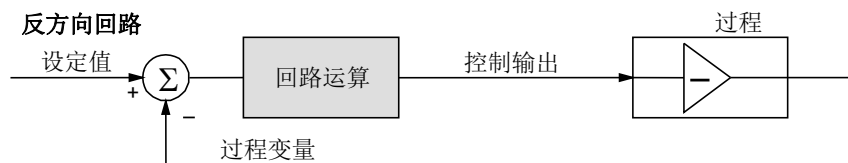
三、正方向和反方向回路

一个过程的增益部分在一定程度上决定了它如何被控制。下图所示的过程有一个正方向增益。这表示过程变量随着控制输出的增加而增加。当然，一个真正的过程通常是包括了时间延迟的复杂的传递函数。这里，我们将关注与控制输出变化相对应的过程变量的变化方向。

大多数过程回路是正方向的，如温度控制。加热会增加PV(温度)，因此，正方向控制有时称为加热型控制。



如下所示，“反方向”控制回路具有负增益。控制输出的增加导致PV的减少。这通常出现在冷却控制系统中，冷却输入的增加会导致PV(温度)的降低。因此，反方向控制有时称为制冷型控制。

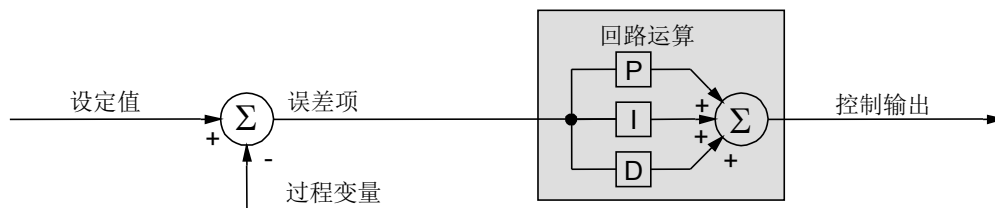


首先，要知道具体的控制回路是正向的还是反向的。如果不是温度控制，就没有明确的答案。在流量控制中，无论是正向还是反向，阀门定位电路都可以很容易地配置并接线。要弄清楚回路方向的最简单方法，就是在手动模式下运行回路。在手动模式下，必须手动产生控制输出值。观察PV随着控制输出的增加是上升了还是下降了。

要进行自动或串级控制，必须正确设置控制输出。(参见上一节有关控制输出设置的内容。)正向控制使用“常规输出”，反向控制使用“反向输出”。为实施反向控制，PID控制器必须使用反向输出。如果PID控制回路构成正方向，在PID控制整定时，对控制数据进行解析。

四、比例、积分、微分项

PID控制方程包括位置型和速度型。方程基本由PID三种基本运算组成。下图是PID计算的概要。控制输出是比例、积分和微分项的总和。每次PID运算时，每一项都会得到同样的误差信号值。



比例、积分和微分项在控制中的作用：

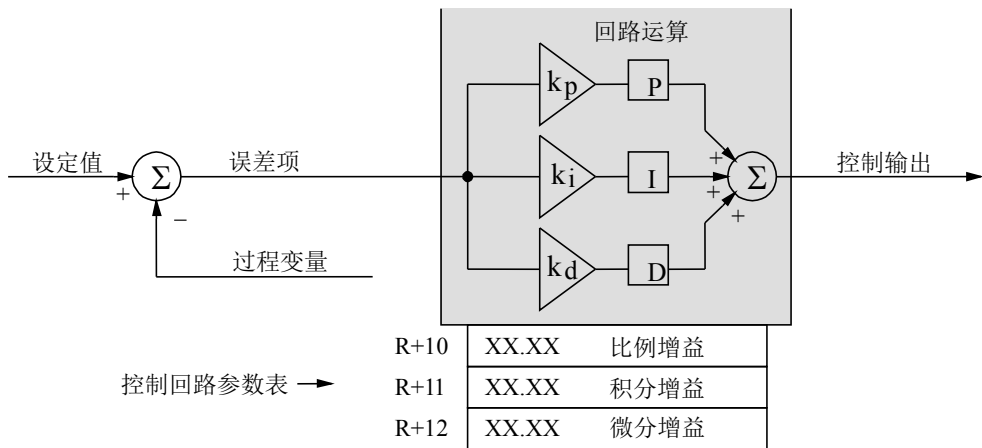
- **比例 (P)**：比例项与当前误差的大小成正比。每次PID运算时，回路控制器都计算出比例项。当误差为零时，比例项也是零。
- **积分 (I)**：积分项对误差值进行积分(求和)。自动模式时，开始PID计算，积分项为每次运算时误差的累加值。使用PID控制位置算法时，如果控制回路已经稳定并且没有误差，则积分项保持PV的当前位置所必需的恒定输出。
- **微分 (D)**：微分项反映了当前误差值与前一次PID计算中使用的误差值的差值。微分项的作用是预计误差会增加并预先对输出产生作用。

比例、积分和微分一般配合使用。为了使PID运算有效果，需要一些额外的指令。P、I和D

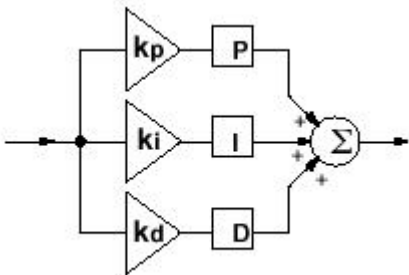
三项分别包含可编程增益值Kp、Ki和Kd，这些值在控制回路参数表位置如下图所示。PID控制调整的目标是得出这些增益值，使控制性能最佳。



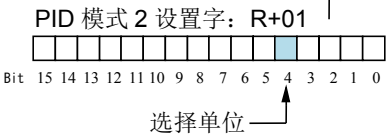
注意：在PID控制术语中，比例增益可以简称“增益”。



P、I和D增益是四位BCD码，数值范围0000～9999。数字在中间包含小数点，所以实际值为00.00～99.99。有些增益有单位，可以通过下面位的设置来指定以秒或分钟为单位的积分增益。微分增益是秒。



R+10	XX.XX	P 增益	—
R+11	XX.XX	I 增益	0=秒、1=分
R+12	XX.XX	D 增益	秒



KPP的PID监控，可以在PID控制运行时，实时设置增益和单位。通常，在PID控制调整期间完成。

比例增益(Kp)：这是三者中最基本的增益。数值范围为0001～9999，内部用作xx.xx。该值设置为“0000”，可去掉比例项，用于ID算法或I算法。

积分增益(Ki)：数值范围0001～9998，内部用作xx.xx。设置为“0000”或“9999”，将使积分时间为“∞”，可去掉积分项，用于比例的应用。如上所示，积分增益的单位可以是秒或分。

微分增益(Kd)：数值范围0001～9999，内部用作xx.xx。设置为“0000”，可去掉微分项，用于PI或I算法。微分项可选择增益限幅特性，详见下一节。



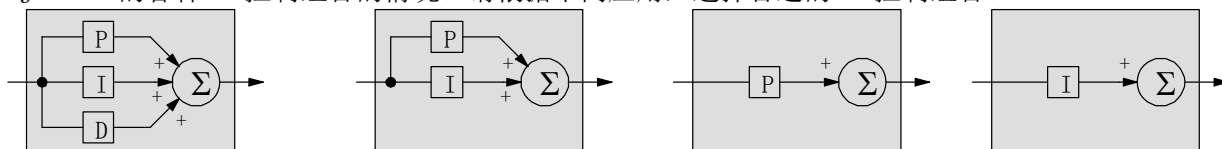
注意：知道如何增加增益非常重要。比例和微分增益，数值越小，产生作用越少；数值越大，产生作用越大。积分作用是增益的倒数(1/Ts)，数值越小，积分作用越大；数值越大，积分作用越小。了解这些知识，对PID控制调整非常重要。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

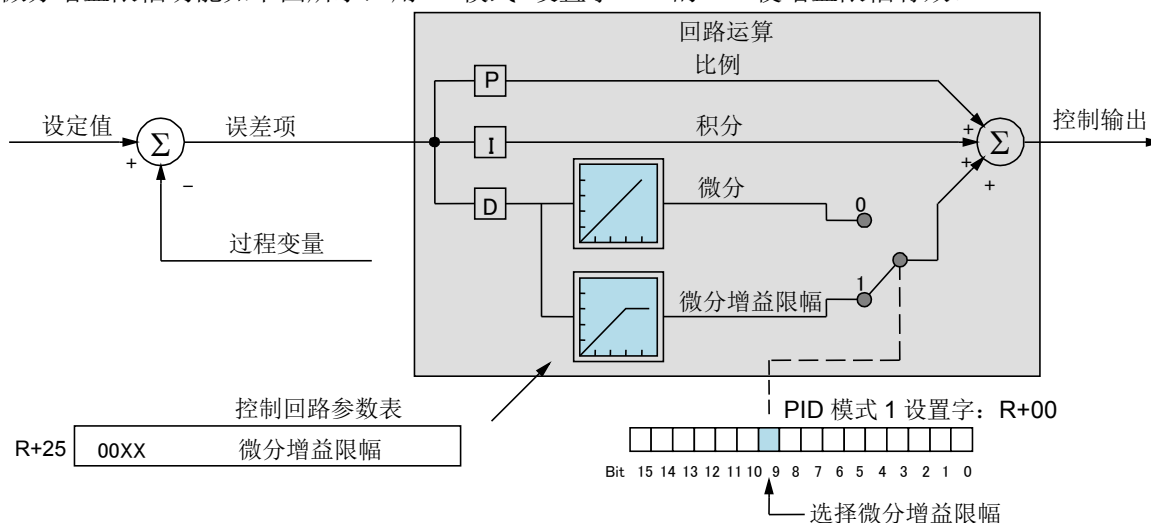
五、部分使用 PID 控制

P、I、D作用都可以从PID公式中去掉。许多应用使用部分PID控制，控制效果更好。下图是SJ-Ether的各种PID控制组合的情况。请根据不同应用，选择合适的PID控制组合。



六、微分增益限幅

这是仅对微分项限幅的功能。选择此功能，使微分项对PV信号的干扰或PV突然波动起作用。微分增益限幅功能如下图所示，用PID模式1设置字R+00的Bit9使增益限幅有效。



在R+25中的微分增益限幅值是一个范围0~20之间的BCD数。只有当Bit9=1时，该设置值有效。微分增益限幅在回路调整中尤为有用。

七、偏差项

在PID控制中广泛使用的位置算法，如下所示的偏差项是控制输出值的重要部分。偏差项在控制回路参数表中的地址是R+04。在每次回路运算后，回路控制器存入一个新的偏差项。

$$M_n = K_c * e_n + K_i * \sum_{i=1}^n e_i + K_r * (e_n - e_{n-1}) + M_o$$

控制输出

比例项

积分项

微分项

初始输出

R+04

XXXX 偏差项

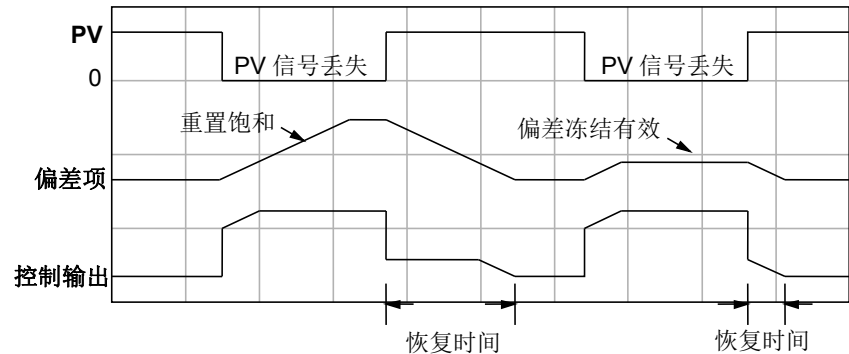
偏差项

如果两个或两个以上的采样周期的误差值(e_n)都为零，则比例和微分项就没有了。偏差项就是积分项和初始输出(M_o)之和。这表示控制输出值为常数，类似于复杂信号的直流分量。

偏差项为控制输出创建一个“有效范围”。当误差在零点附近波动时，输出就会在偏差项附近波动。这个概念非常重要，它说明了为什么积分项对误差的响应必须比微分项或比例项慢得多的原因。

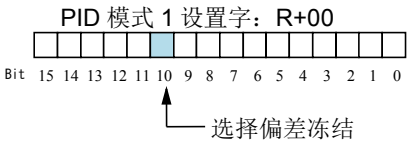
八、偏差冻结（积分分离）

发生“重置饱和”后，积分运算会产生明显的过冲，参见下图。假设PV信号断开，PV值将为零。在严重的PID控制故障期间，重置饱和和将会变得更糟。请注意，当PV信号断开时，偏差项将累加到达到上限值。当PV信号恢复后，偏差项处于饱和，需要很长时间才能恢复到正常值，导致控制输出恢复时间延长。在恢复之前，控制输出电平将不正确，导致问题。



在上图例子中，第二次PV信号断开时，偏差冻结功能有效。控制输出超出限幅时，对偏差项进行冻结。这种方式可以避免重置饱和，控制输出恢复时间要少得多。

在大多数应用中，偏差冻结功能与PID控制一起使用。使用KPP的PID设置对话框，或如右图所示将PID模式1设置字的Bit10置位，启用偏差冻结功能。



注意：当控制输出达到极限值时，偏差冻结功能停止偏置项的作用。如果控制输出的设置范围超出输出限制(单极性/12位，范围0-4095，设定在4046以上)，偏差冻结功能将不起作用。

在本章稍后将描述的前馈中，偏差项可以通过梯形图程序直接写入。但是，这与偏差冻结功能并没有冲突，因为在控制回路运行中偏差条件很少通过前馈写偏差项。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

第八节 PID回路整定方法

回路整定可能是闭环过程控制中最重要的一步，其目的是通过调整回路增益，使得回路可以在各种条件下获得最佳性能。一般来说，通过改变SP后PV能否很好地追踪SP，判断控制性能的质量。

自整定和手动整定：可以直接改变PID增益值(手动整定)，也可以通过CPU内置PID处理引擎自动计算增益(自动调整)。自整定计算增益值比手动整定更容易。通过自整定，可以获得接近最佳的增益。通过手动整定进行微调，可以获得最佳的增益值。



警告：只有熟悉过程的人员，才能对回路整定常数进行改动。在控制输出值急剧变化的过程中，自整定会 对过程产生影响。为了尽量减少对人员伤害和设备的损坏，要全面考虑每一项变更的影响，熟悉过程的知识，再使用SJ-Ether进行自整定。

一、开环测试

无论是使用手动整定还是自整定，在整定之前，确认新安装过程的基本特征非常重要。在手动模式下，确认新回路的以下参数：

- **设定值：**确认设定值数据源是否正确。将PLC置于运行模式，回路处于手动模式。监视控制回路参数表R+02，查看SP的值。同时确认上升/保持发生器(如果使用)。
- **过程变量：**PV值在控制回路参数表R+03。监控R+03，确认测量值准确。如果PV信号经常波动，可以通过硬件(RC低通滤波器)，或使用数字式S/W滤波器对PV信号进行滤波。
- **控制输出：**在保证安全的情况下，手动将输出值作较小的改变(约10%)，观察对过程变量的影响。弄清楚过程是正方向还是反方向，并检查控制输出的设置(正方向或反方向)。确保控制输出的上限和下限不相等。
- **采样周期：**在开环状态下，可以找到理想的采样周期(其步骤在本章的前面已介绍)。但是，如果使用自整定的话，除了PID增益之外，自整定过程还将自动计算采样周期。

下一页开始讨论手动整定方法。如果只想进行自整定的话，请跳过下一段，直接进入自整定那一段。

二、手动整定方法

在切换到自动方式前，先检查以下项目：

- 用附属设备监控回路参数，推荐使用KPP软件的PID视图功能。



注意：使用PID趋势图设置菜单，可以把SP/PV和偏差/控制输出等区域设置成横轴。但自动缩放功能会通过别的方法，把这些过程参数改变为纵坐标显示，从而造成回路整定过程的混乱。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

- 使比例增益=10，积分增益=9999，微分增益=0000。这使积分项和微分项无效，而提比例增益较小。

- 检查控制回路参数表R+04中的偏差项值，如果不是0，则使用编程软件将0写入R+04。现在可以把回路切换到自动模式，检查模式监视位，确认当前模式。如果没有转换到自动模式，请参考故障处理一节。



警告：如果PV值和控制输出值开始振荡，请立即减小比例增益，如果PID回路不稳定，则返回到手动模式，手动将控制输出设置为安全值。在PID回路整定过程中，人要在急停开关附近，以便随时停止回路运行。

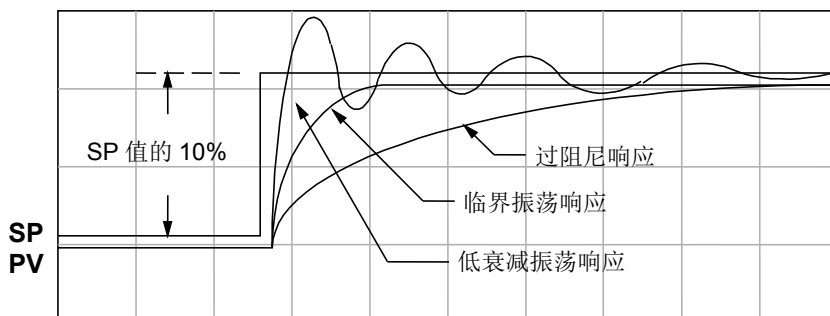
- 因采用无扰动切换功能， $SP = PV$ 。增大一点SP，以产生一个误差值，仅使用比例增益，并使偏差项=0，就可以容易地检查控制输出值：

$$\text{控制输出} = (SP - PV) \times \text{比例增益}$$

- 如果控制输出值改变，执行机构、加热器或其它设备就会作用于回路，PV不久就会接近SP值。如果PV不改变，则增大比例增益直到它有变化。
- 现在，增加一点的积分增益。要记住，数值越大则积分增益越小，数值越小则积分增益越大！经过这一步后，PV应当等于SP或非常接近SP。

到目前为止，我们仅使用了比例和积分增益。接下来，我们将SP改变10%，给过程一个冲击，调节增益值，得到PV的最佳响应。参考下图，根据KPP软件的PID趋势图调节增益。在没有振荡的情况下，找出PV的最快响应。

- 过阻尼响应：增益太小，要逐渐增大比例增益。
- 低衰减振荡响应：增益太大，先减小积分增益，必要时减小比例增益。
- 临界振荡响应：这是最佳增益设定值，可以稍微增加比例增益，确认是否有1~2次小波动。



现在可以增加一点微分增益，进一步改善上述的临界振荡响应。注意，此时比例和积分增益非常接近最终值。在不引起回路振荡的情况下，增加一些微分作用，会增加比例增益。微分作用使比例作用稍微平滑，所以要一起调整这些增益。

三、自整定方法

使用KPP进行自整定。首先，使用自整定来设置PID参数(常量)值，但自整定不会连续运行。每当控制回路发生相当大的变化时(过程量、执行机构尺寸等)，都需要再次自整定，以获得最佳控制所需的新增益。



警告：只有熟悉过程的人员，才能对回路整定常数进行改动。在控制输出值急剧变化的过程中，自整定会 对过程产生影响。为了尽量减少对人员伤害和设备的损坏，要全面考虑每一项变更的影响，熟悉过程的知识，再使用SJ-Ether进行自整定。

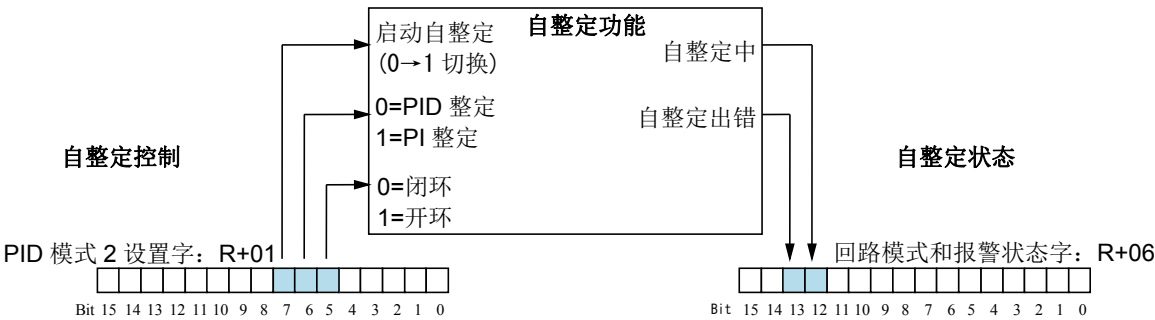
注意：
确保



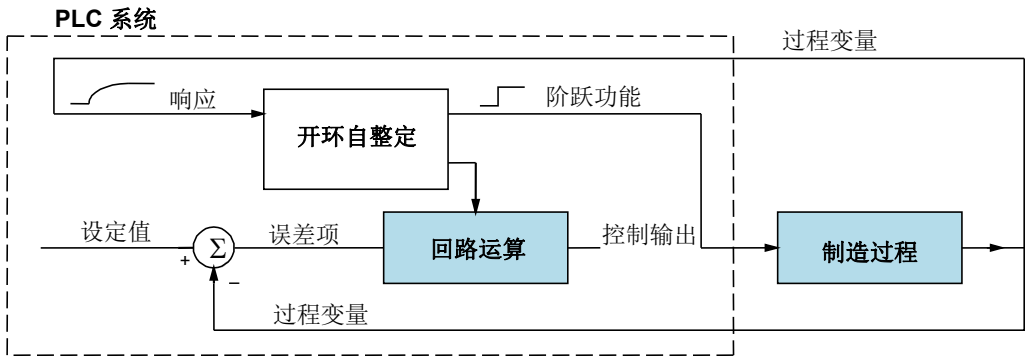
DirectSOFT4、DirectSOFT5不支持SJ-Ether。
使用KPP Ver1.6.1.2或更高版本。

回路控制器有开环和闭环两种方式。使用手动模式自整定时，控制不能在自动模式下停止，这时请先使用开环方式。当回路在自动模式下时，允许使用自整定的闭环方法。下一节说明了如何使用自整定功能，以及在开环和闭环自整定时 的情况。

如下所示，自整定控制功能使用了PID模式2设置字R+01中的三个位。在KPP软件中使用自整定功能时，会自动处理这些位。通过设置各个控制位，可以启动自整定过程，可以选择PID或PI整定，可以选择开环或闭环整定。如果选择PI整定，自整定将把微分增益置为0。回路模式和报警状态字R+06显示自整定的状态。当自整定期间，Bit12位将为on（1），当自整定结束后，它自动回到off（0）。



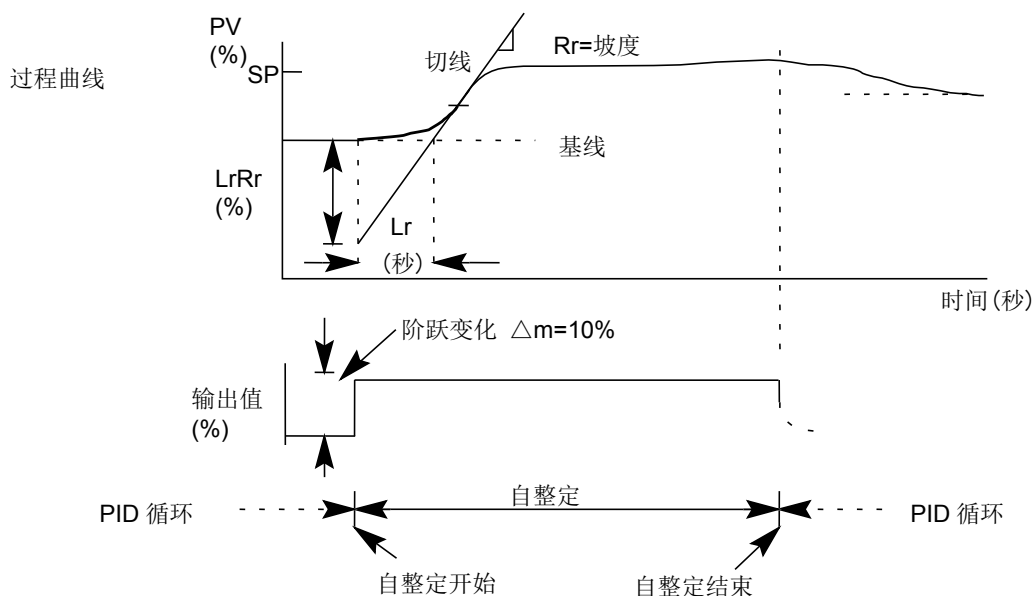
开环自整定：在开环自整定期间，回路控制器按下图所示工作。在开始整定前，先使回路处于手动模式，保证PV和控制输出值在限幅范围的中间位置(远离极限值)。



注意：由于不是闭环回路，SP的值无关紧要。但是，固件要求在开始自整定前，SP的数值应比PV数值大205（正作用时SP大于等于205，反作用时SP小于等于205）。

自整定时，回路控制器会使输出产生一个阶跃变化，观察PV的响应，自整定功能根据PV响应，计算出增益和采样时间。它自动地将结果存放在控制回路参数表相应的寄存器中。

下面的时序图显示了开环自整定的结果。自整定功能使控制输出并产生10%的阶跃变化。如果回路控制器观察到的PV变化小于2%，那么将输出阶跃变化增加至20%。



- 自整定开始时，阶跃变化输出 $\Delta m=10\%$ 。
- 自整定时，控制器输出达到满刻度正极限值。自整定停止，在自整定错误标志位被置位。
- 当PV的变化小于2%时，使输出产生20%的变化。

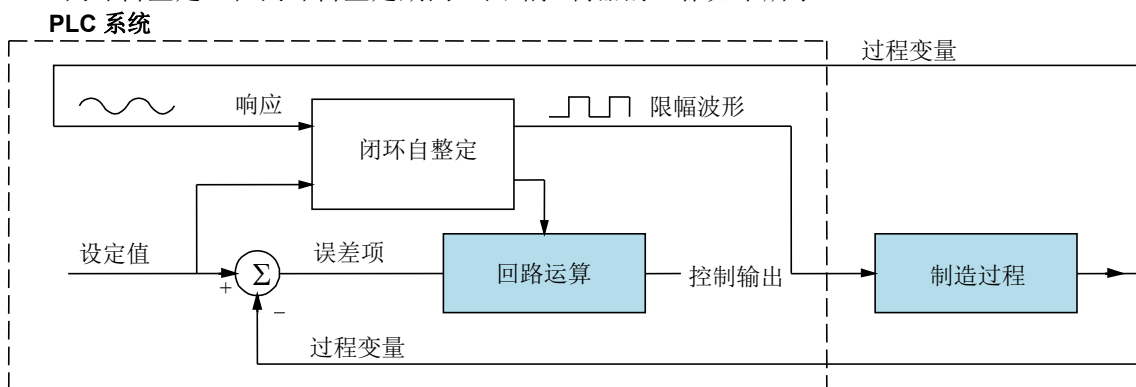
当回路调整监控结束后，回路控制器计算出 R_r (最大斜率，以%/秒为单位) 和 L_r (静止时间，以秒为单位)。自整定功能按齐格勒-尼古拉斯公式计算出增益，如下所示：

PID 整定	PI 整定
$P = 1.2 \times \Delta m / L_r R_r$	$P = 0.9 \times \Delta m / L_r R_r$
$I = 2.0 \times L_r$	$I = 3.33 \times L_r$
$D = 0.5 \times L_r$	$D = 0$
采样周期 = $0.056 \times L_r$	采样周期 = $0.12 \times L_r$
$\Delta m = \text{输出阶跃变化 (10\% = 0.1、20\% = 0.2)}$	

我们强烈建议使用KPP进行自整定。每次自整定的持续时间取决于过程量。PV变化缓慢，则需要更长的自整定时间。自整定完成后，将比例、积分和微分增益值，分别自动更新到控制回路参数表的R+10、R+11、R+12。R+07的采样时间也会自动更新。可以通过测量闭环下PV对输出中阶跃变化的响应，来测试自整定是否有效。方法参见手动整定方法一节。

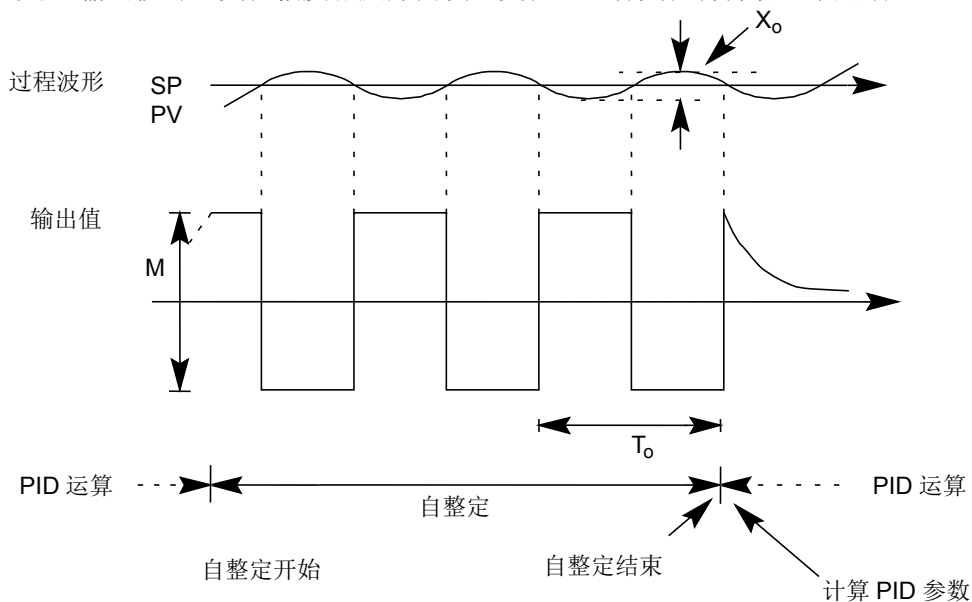
自整定错误：如果自整定错误位(控制模式和报警状态字R+06的位13)置位，确认PV值和SP值是否在满量程的5%以内(自整定的运行条件)。如果使用闭环方法，输出超出范围，该位也会置位。

闭环自整定：在闭环自整定期间，回路控制器的工作如下所示。



自整定时，回路控制器将在控制输出发出方波。当PV值超过(或低于)SP值时，输出波形将改变。因此，限幅波形的周期大致与过程对象的变化次数成正比。PV开始响应后，自整定功能计算出增益和采样时间，自动地将结果存放在控制回路参数表相应的寄存器中。

下面的时序图显示了闭环回路自整定的过程。自整定功能检查PV与SP的差值的正负号，然后对控制输出进行控制，产生与误差值符号相反的满幅度阶跃变化。每次当误差SP-PV的符号发生改变时，输出按与PV变化幅度相反方向发生变化，这种变化将持续三个周期。



*Mmax=输出值的上限设定值 Mmin=输出值的下限设定值

*此例为正方向。当设为反方向时，输出相反。

结束回路调整监控完成以后，回路控制器计算出 T_o (振荡周期)和 X_o (PV的振幅)，然后用这些值计算出 K_{pc} (灵敏度限幅)和 T_{pc} (周期限幅)。从这些值中，回路控制器的自整定功能按如下所示的齐格勒-尼古拉斯公式计算出PID增益和采样周期：

$K_{pc}=4M / (\pi \times X_o)$	$T_{pc}=0$
M=输出振幅	
PID 整定	PI 整定
$P=0.45 \times K_{pc}$	$P=0.30 \times K_{pc}$
$I=0.60 \times T_{pc}$	$I=1.00 \times T_{pc}$
$D=0.10 \times T_{pc}$	$D=0$
采样周期 $=0.014 \times T_{pc}$	采样周期 $=0.03 \times T_{pc}$

自整定错误：如果自整定错误位(控制模式和报警状态字R+06的位13)置位，确认PV值和SP值是否在满量程的5%以内(自整定的运行条件)。如果使用闭环方法，输出超出范围，该位也会置位。



注意：如果PV波动较大，则利用梯形图程序进行滤波。

四、串级回路的整定

串级回路调整中，首先断开串级方式，用前述PID控制的调整方法对回路逐个调整。

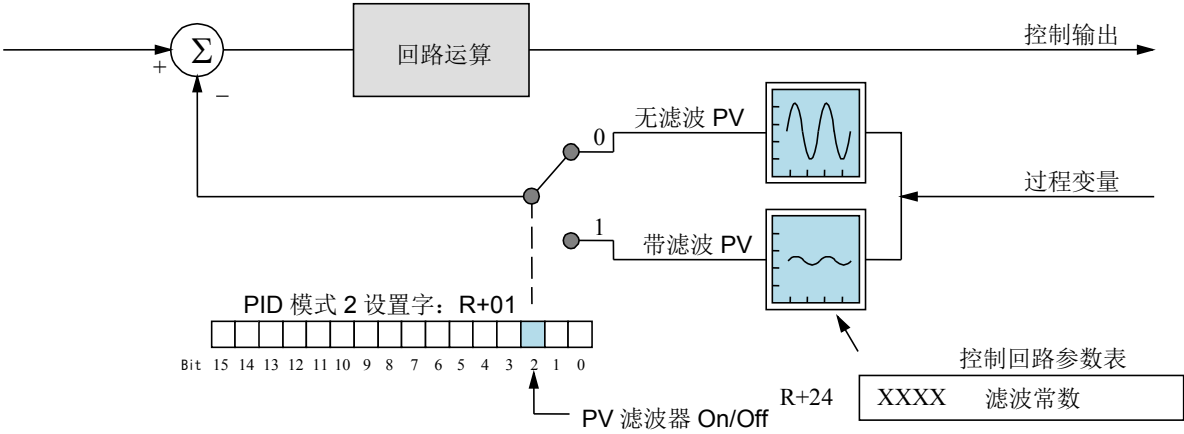
1. 当使用自整定时，根据本章描述的方法，先找到副环的采样周期。随后，设置主环的采样周期。由于10个因素，主环的采样周期比副环的慢。以此为自整定的起点。
2. 先调整副环，将主环切换到手动模式，按上述所说的回路调整方法，手动改变副环的SP值。
3. 使副回路处于自动模式，在SP变化10%时，副环处于临界振荡。这样就完成了副环的调整。
4. 这一步中，使副环处于串级模式，主环处于自动模式，进行整定。将副环作为过程的一个串联部分，调整主环。调整主环后，再调整副环。
5. 在本节中，根据标准PID回路调整方法，调整主环。主环的PV响应是整个串级PID控制的响应。

五、在程序中加入模拟量滤波

带有噪声的PV信号使整定变得困难，输出试图响应PV的波峰和波谷，所以控制输出偏离实际值。为使PID控制稳定，可以对P V 输入进行滤波。使用梯形程序，可以得到相同的效果。

PV自动传输功能内置模拟滤波器

可以选择PV自动传输功能的PV输入滤波器，对模拟量输入(PV值)进行滤波。当使用闭环自整定时，特别有用。如下图所示，建议在自整定时使用此滤波器。自整定结束后，可以禁用滤波器。如果PV输入信号波动，则继续使用滤波器。



PID模式2设置字的Bit2是PV输入滤波器的设置位(0=无效, 1=有效)。使用控制回路参数表的寄存器R+24设置滤波器常数。如下所示，滤波器常数值的数据格式为BCD，表示小数点00X. X。

我们强烈建议使用KPP进行自整定。每次自整定的持续时间取决于过程量。PV变化缓慢，则需要更长的自整定时间。自整定完成后，将比例、积分和微分增益值，分别自动更新到控制回路参数表的R+10、R+11、R+12。R+07的采样时间也会自动更新。可以通过测量闭环下PV对输出中阶跃变化的响应，来测试自整定是否有效。方法参见手动整定方法一节。

- 内置滤波器采用以下算法：
- $Y_i = k (X_i - Y_{i-1}) + Y_{i-1}$
 - Yi：滤波器的当前输出
 - Xi：滤波器的当前输入
 - Yi-1：滤波器的上一次输出
 - k：PV模拟量输入滤波器系数



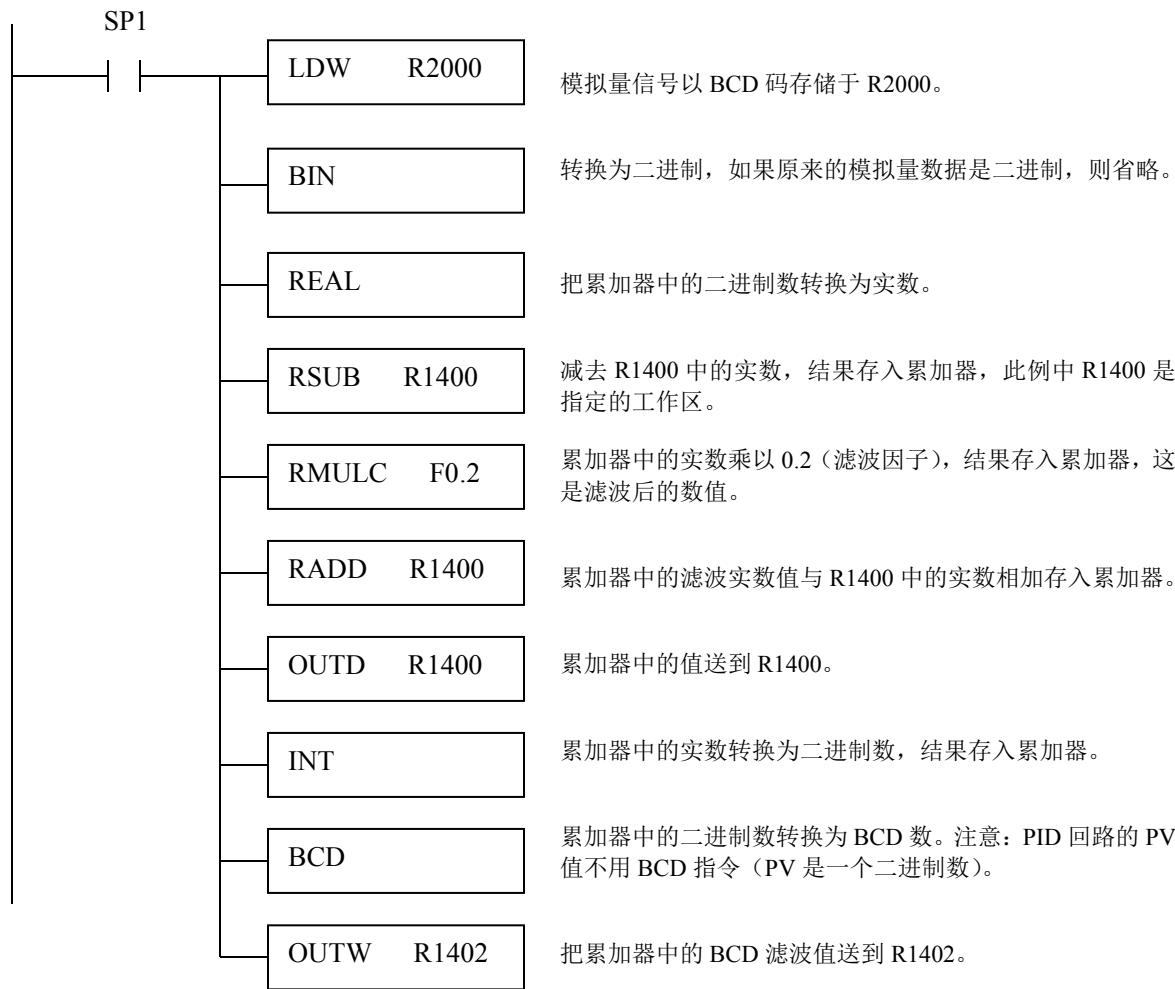
注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。



注意：请将模拟量滤波功能与PV自动传输功能一起使用。

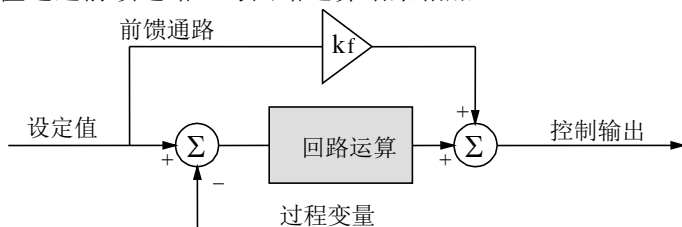
为了去除从模拟量输入端进来的干扰信号，需要对模拟量输入进行滤波，下面是滤波程序的例子，可以根据自己的应用情况改变例子程序中的寄存器。

因“四舍五入”的原因，滤波会引起小的误差，由于可能的化整误差，请不要使用0或满量程刻度作报警点，另外，滤波常数越小，则滤波效果越好，但会减慢响应时间，要确保控制过程的最慢响应速度在可接受的范围内。



第九节 前馈控制

前馈控制比标准闭环控制更为优化。它对PID控制中的不稳定性或设定值的突然改变非常有用。虽然前馈控制是很好的功能，但前馈控制对于回路调整不是最好的调节手段，一般情况下请使用标准闭环PID控制，只有在需要进一步提高控制性能时才使用它。“前馈控制”的术语，参见下图。设定值通过前馈通路，与回路运算结果相加。

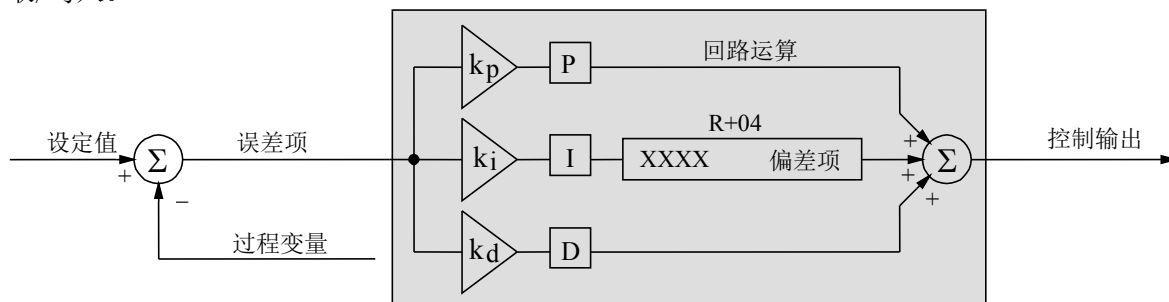


在上一节偏差项的介绍中，说明了偏差值是成为控制输出的工作点，误差在零点附近波动，输出也以偏差值为中心变化。当设定值发生变化时，会产生误差，控制输出也产生新的工作点。如果有干扰，会在PID控制中产生新的偏移量。积分器（偏差）也随之增加或减少，直到没有误差，偏差项运行于新的工作点。

假设设定值发生突然发生变化（在应用中很常见），如果能快速将输出变更为新的工作点，使输出尽快达到新工作点，则可以避免由设定点更改导致的误差。在更改设定点后，如果知道工作点（偏差值）最佳值，则可以直接手动更改输出，这就是前馈控制。前馈控制具有以下优点：

- 只要可以预测导致PID控制设定点和偏移量变化的干扰，就可以减少SP-PV错误。
- 正确使用前馈控制，可以降低积分增益，使控制系统更稳定。

如下所示，前馈控制器非常易于使用。偏差项位于控制回路参数表(R+04)中，可由用户读取/写入。



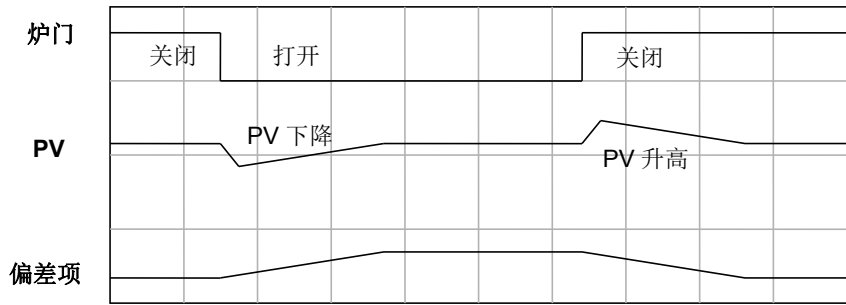
要改变偏差项（工作点），通过梯形图程序，将所需值写入R+04。PID控制回路首先读取R+04中的偏差项，并根据当前的积分运算修正该值，最后将结果写入R+04。前馈控制唯一需要做的就是适当的时间，写入合适的偏差项值。



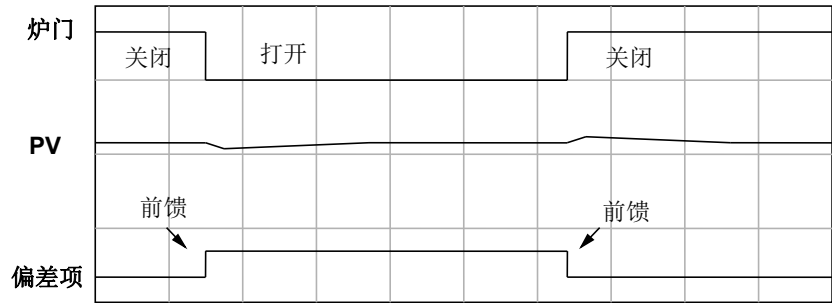
注意：在写入偏差项值时，必须设计梯形图程序，在产生新偏差项工作点的时刻只写入一次值。如果每次扫描，梯形图程序都写入偏差项值，则PID控制的积分不起作用。

前馈举例

我们将解释何时写偏差值和写入值。例如，假设有炉温控制回路，已经进行PID整定，并取得了最佳性能。参见下图，当操作员打开炉门时，PID控制偏差项适应热量损失，温度稍微下降。炉门关闭时，直到回路再次调整前，温度会上升到SP以上。前馈控制可以抑制这种影响。



首先，记录开门和关门时回路控制器产生的偏差变化量。接下来，编写一段梯形图程序来监控炉门的限位开关位置。门打开时，梯形图程序从R+04读取当前的偏差项值，加上必要的变化量，再将其写入R+04。门关闭时，减去刚才的变化量。结果如下图所示。



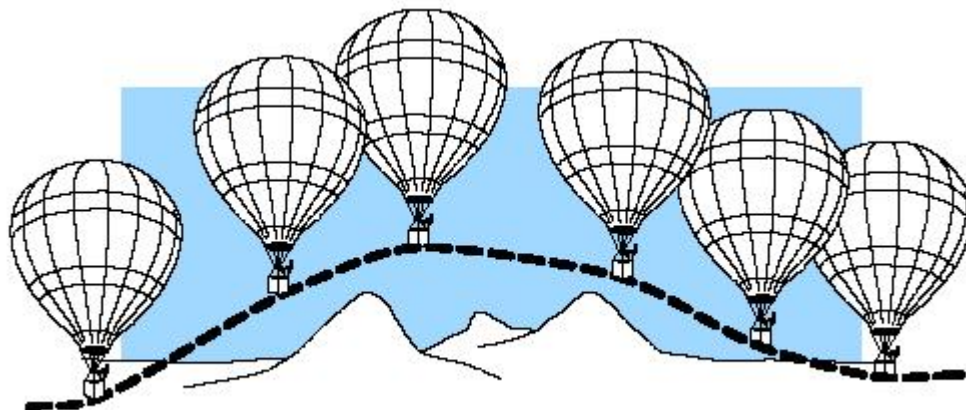
偏差项的阶跃变化是对偏差项值两次前馈写入的结果。可以看到PV波动被显著减少，同样的方法也可用于设定值的改变。

第十节 时间-比例控制

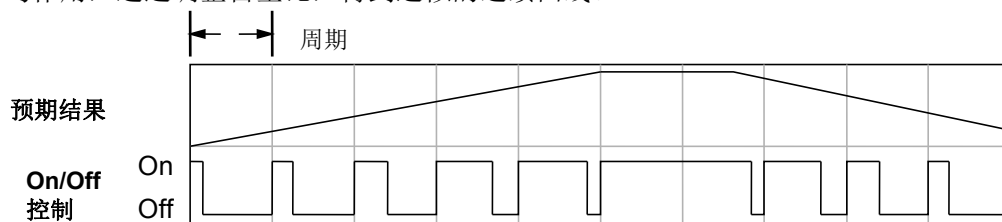
PID控制产生一个平滑的控制输出信号。控制输出值适用于驱动与过程连接的模拟量输出模块。由于输出是持续的，在过程控制领域称为连续控制。

要使用连续控制，使输出平稳且达到高精度，PID控制回路的部品(如执行机构、加热器放大器等)的成本很高。有一种更简单的控制称为时间-比例控制。此方法控制执行机构开启或关闭(无中间状态)。基于开/关控制的执行机构比连续控制的执行机构更便宜。

本节介绍如何将PID控制的控制输出转换为时间-比例控制。以下说明通过开启和关闭负载来控制过程的机制。下图显示了一个热气球如何越过几座山脉。此情况下，路径是设定值。气球飞行员反复开关燃烧器，这是控制输出。气球中的大量空气有效地平均了燃烧器的效果，使热量连续变化。缓慢变化的气球温度及其高度是过程变量。



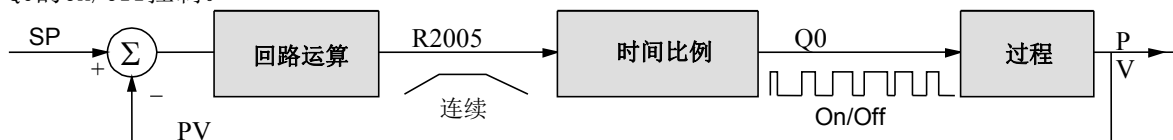
时间-比例控制通过调节占空比(开与关的时间比率)近似于连续控制。下图显示由于大量过程的平均作用，通过调整占空比，得到近似的连续曲线。



如果记录热气球燃烧器的开/关时间，可以看到燃烧器对气球的温度和高度影响有类似关系。

On/Off控制的程序例子

下面梯形图程序按时间比例控制输出的On/Off。它把R2005中的连续输出值转换成输出线圈Q0的On/Off控制。



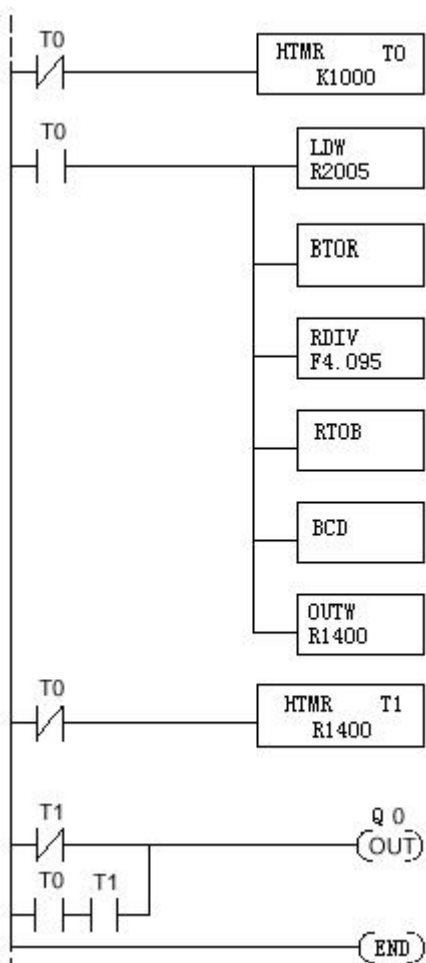
两个定时器用于开/关控制，可以根据应用的需要进行调整。在时间比例程序中，T0的时间分辨率要与输出的分辨率相匹配。

- 控制回路参数表以R2000为起始地址，控制输出存放于R2005。
- 控制输出的数据格式为12位，无符号(0~FFF或0~4095)。
- 开/关控制的输出是Q0。

开/关波形的时基(一个完整周期)为10秒。使用高速定时器，以0.01秒为增量计数1000(10秒)。



注意：某些过程太快，不适合使用时间-比例控制。选择这种控制方法时，要考虑用户过程的速度。过程变更速度过快，不适合时间-比例控制，请使用连续控制。



使用0.01秒定时器, K1000为10秒预置值, T0常闭触点使其自复位, T0触点每10秒接通一个扫描周期。

到10秒, T0接通, 从控制回路参数表R+05 (R2005) 读入控制输出值(二进制)

因为要作无量纲换算, 所以将数值转换为实数

用控制输出除以4.095, 将0~4095之间的数值转换为0~1000范围的数值, 和10秒计时器范围相匹配

将实数转换为二进制数

将累加器中的数值转换为BCD码

结果输出到R1400, 作为第二个定时器预置值

预置值存放于R1400, 最大10秒定时, 当达到预置值时, Q0由On变为Off

输出线圈在10秒时基开始时On

主程序结束

第十一节 串级控制

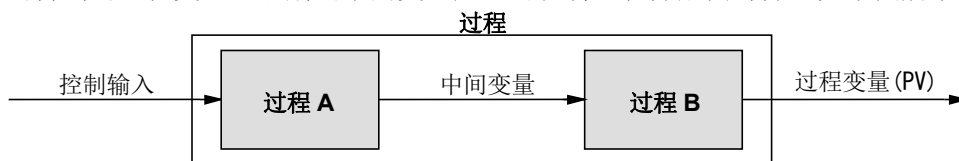
一、简介

在某些情况下，串级控制是一种先进的控制技术。顾名思义，串级就是一条回路连接到另一条回路。除了手动(开环)和自动(闭环)模式外，SJ-Ether还有串级模式。



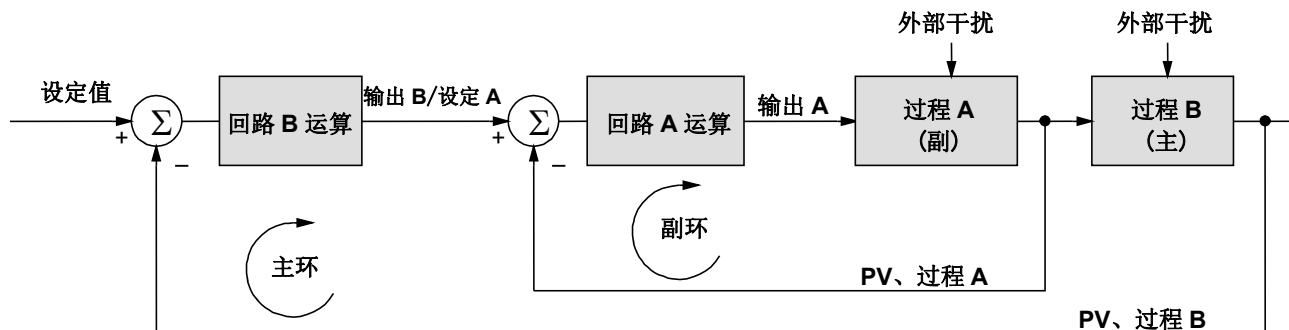
注意：串级回路是一种先进的过程控制技术。所以建议只有富有经验的过程控制工程师才能使用它。

如果制造过程复杂且从控制输入到过程变量输出存在时间滞后，则再怎么调整的单个回路，还是对速度和精度控制不足。执行机构对某些物理特性的产生影响，最终影响对不同物理特性测量而得到的过程变量。识别出中间变量后，可以将过程分成两部分，如下图所示。



串级控制的原理很简单，可以通过添加另一个过程控制来更精确地控制中间变量。另外，这种方法可以将控制延迟的原因分成两部分。

下图是一个串级控制系统，只是使一个PID回路嵌套到另一个PID回路。内部回路称为副环，外部回路称为主环。为了整体的稳定性，副环应该比主环响应得更快。另外，还需要添加传感器，测量中间变量(过程A的PV)。注意，副环的设定值是由主环的输出自动产生的。对串级控制进行编程和调试后，只需在系统级别处理初始设定值和过程变量。串级控制可以看成是一个回路，但改善以前单回路的性能。



通过检查系统对外部干扰的响应，可以看到串级控制的优越性。副环比主环响应快。因此，如果副环的过程A受到干扰的影响，则主环受影响之前，通过回路A的PID运算来修正误差。

二、SJ-Ether CPU 的串级回路

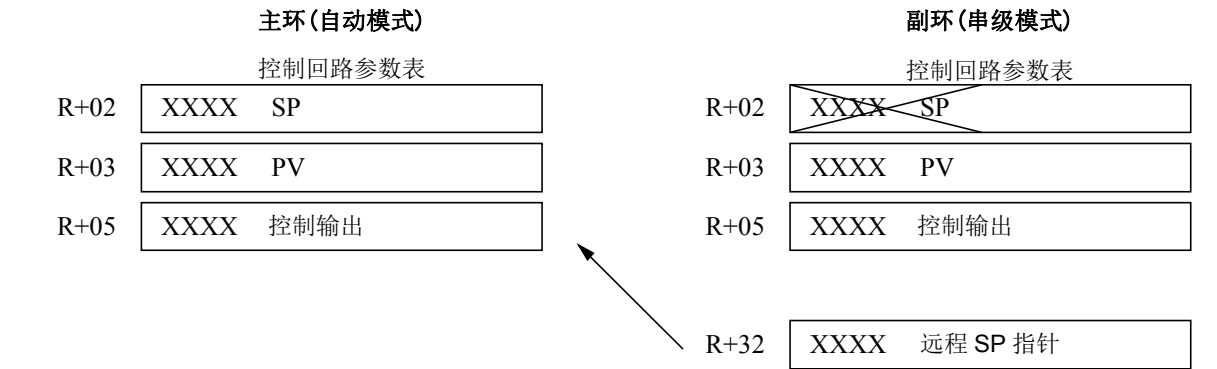
在使用“串级回路”时，必须进行重要区分。实际上，只是副环处于串级模式。在正常运行中，主环必须设置为自动模式。当多个回路串级时，通常最外层(主环)回路设置为自动模式，将最内层回路设置为串级模式。



注意：按严格的过程控制术语，主环和副环都是“串联”的。遗憾的是，在控制中我们无法恪守此规定。只要记住，副环处于串级模式，主环处于自动模式。

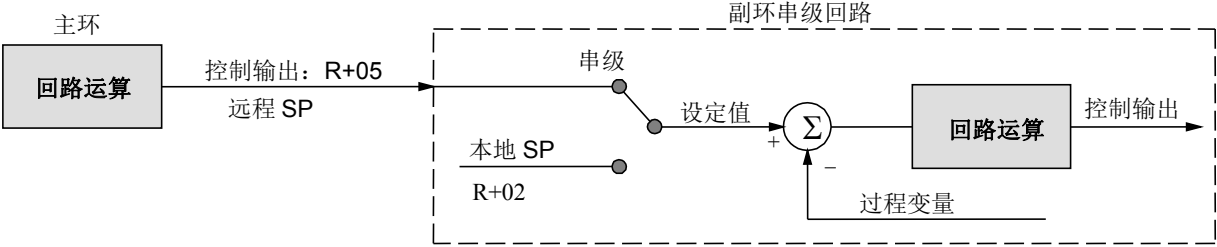
使用SJ-Ether，可以将多个PID回路串联在一起，可以有多个串级回路。为了正确控制串级回路，主环和副环回路必须设置相同的数据范围(12/15位)及相同的符号(单极性/双极性)。

为使副环在串级模式下运行，必须用梯形图程序将远程设定值的指针存放到控制回路参数表中的R+32。指针必须是主环控制输出的地址(R+05)。在串级模式下，副环忽略其本地SP值(R+02)，读取主环的控制输出作为副环的SP值。



如果使用KPP的PID视图监控副环的SP值，KPP会自动读取主环的控制输出并将其显示为副环的SP值。副环的正常SP值(R+02)不会改变。

在此，绘制一个与上面回路参数等效的回路图。



如上所述，如果副环退出串级模式，那么主环自动变成手动模式。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。

确保使用KPP Ver1.6.1.2或更高版本。

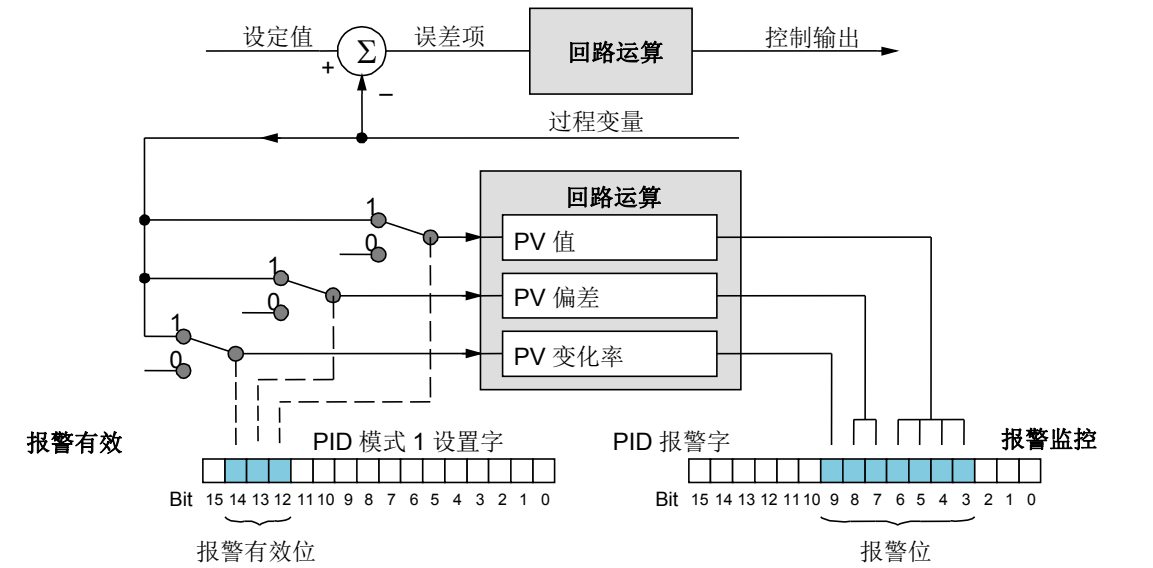
第十二节 过程报警

过程控制的性能通常取决于过程变量与设定值的接近程度。实际运行中，大部分过程控制都是连续工作，当发生错误时，将可能使PV失控。为了及早发现回路的错误，必须要有过程报警。如果发生错误，会发出报警，手动控制回路或者采取其他措施，直到恢复原有状态。

SJ-Ether CPU具有最新的PID控制报警功能。

PV绝对值报警	设置PV两个下限值和两个上限值，并对PV值进行监控。如果PV值超过了设定的限值，则会产生报警。
PV偏差报警	监控PV值与SP值之差。如果PV和SP之间的差值超过编程的报警值，则会产生报警。
PV变化率报警	计算PV的变化率，如果超过了设定的报警值，则产生报警。
报警滞后	与PV绝对值报警和PV偏差报警一起使用，可减少PV在报警阈值附近波动造成报警的次数。报警阈值可以通过编程改变。每个警报都可以独立监控。

下图显示了PV监控功能。通过对控制回路参数表的PID模式1设置字(R+00)的Bit12~14的设置，启用/禁用报警。KPP的PID视图设置对话框可以简化对报警的编程和监控。在梯形图程序中，通过检查控制回路参数表的PID模式和报警状态字R+06的Bit3~9，可以监视报警。



与PID计算不同，只要CPU处于运行模式，报警就会起作用。控制模式可以是手动、自动或串级。如果将上面列出的使能位设置为“1”，则报警起作用。

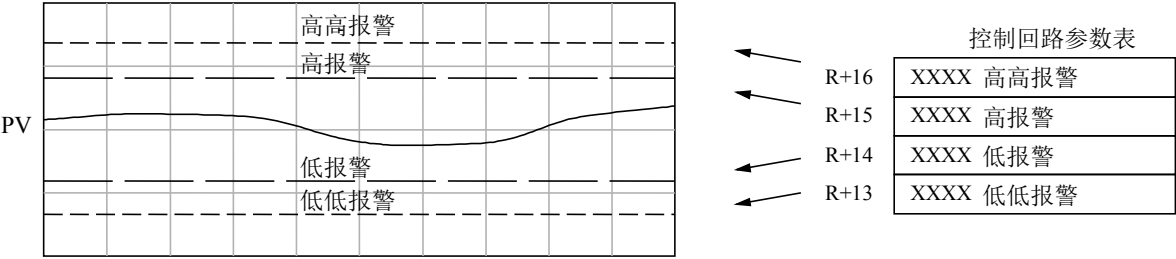


注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

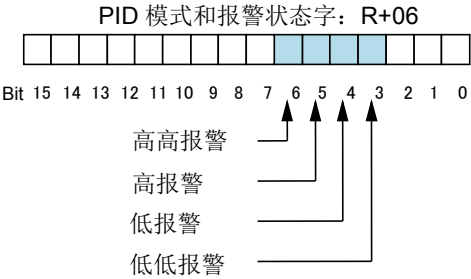
一、PV 绝对值报警

PV绝对值报警由两个上限报警和两个下限报警组成。如下图所示，只要PV值处于上限报警和下限报警之间的范围内，就不会报警。安全区附近的报警称为高报警和低报警。如果回路失控，PV值首先超过这两个阈值之一。因此，可以如下右图所示，在控制回路参数表相应地址设

置报警阈值。PV和SP的数据格式相同(12位或15位)。如果过程失控，则会提早发出警告。



如果过程继续失控，最终PV值超过外部报警阈值，即进入高-高报警或低-低报警的范围。其阈值也在控制回路参数表寄存器设置。高-高报警或低-低报警表示状况危险，需要操作员立即注意。如右图所示，使用控制回路参数表中PID模式和报警状态字的4位通知PV绝对值报警。强烈建议用梯形图程序监控这些位，可以通过位操作指令轻松完成，也可以使用KPP来监控PID报警。

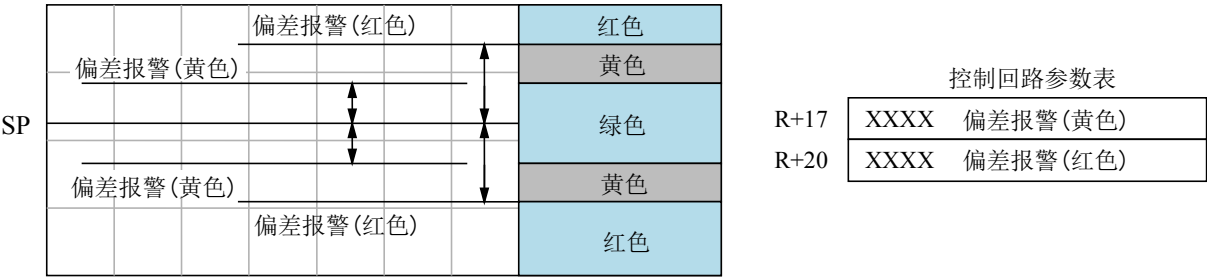




注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

二、PV 偏离报警

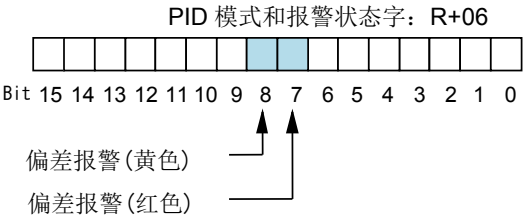
PV偏差报警监控PV与SP的偏差。偏差报警有两个可编程的阈值，每个阈值对称于当前SP值的上端和下端。在下图中，偏差较小的报警称为“黄色偏差”，表示要注意PID回路的状态。偏差较大的报警称为“红色偏差”，表示PID回路的偏离非常严重。如下图所示，在控制回路参数表R+17和R+20设置阈值。



这些阈值定义了SP值的变化区域。SP值的绿色区域表示安全状态(无警报)。黄色区域在绿色区域之外，红色区域在黄色区域之外。

PV偏差报警由控制回路参数表的PID模式和报警状态字中两个Bit表示，如右图所示。我们建议使用梯形图程序来监视这些位。位操作指令可以很容易做到。另外，可以使用KPP监控PID报警。

使用PID模式1设置字R+00的Bit13，PV偏差报警可以独立于其他PV报警而启用或禁用。



报警滞后功能可以与偏差报警和绝对值报警一起配合使用。在本节末尾讨论。

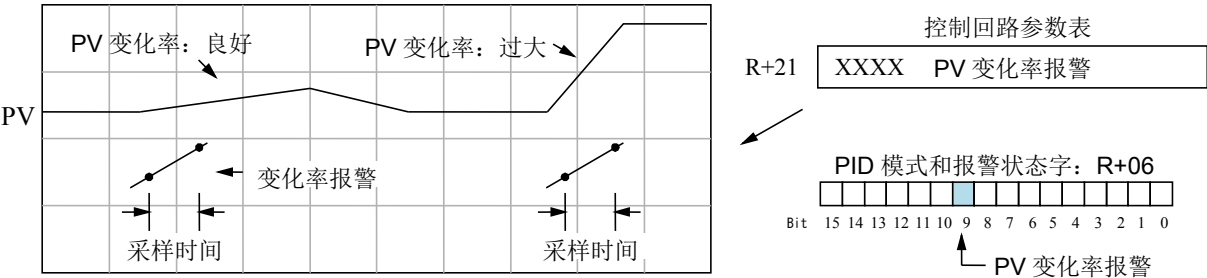


注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

三、PV 变化率报警

要尽早发现过程故障的一种有效方法是监控PV变化率。大多数控制，PV值都缓慢变化。如果PV值的变化变快了，可能是由于PV或控制输出信号线断开、SP值错误等原因造成的。如果操作员快速有效地响应PV变化率报警，PV绝对值将不会使正在处理的物料报废。

PID控制可以如下所示设置PV变化率报警，设定PID回路的每个采样时间的PV变化量。该值可用编程方式在控制回路参数表的R+21。



假定PV是过程温度，如果温度变化超过15度/分钟，则进行报警。这时需要知道每度的PV计数和PID控制采样周期。如果PV值(R+03)为每度10个计数, 回路采样周期为2秒, 则使用以下公式将测量单位转换为计数/采样周期。

变化率报警 = $\frac{15 \text{ 度}}{1 \text{ 分钟}} \times \frac{10 \text{ 计数/度}}{30 \text{ 次采样/分}} = \frac{150}{30} = 5 \text{ 计数/采样周期}$

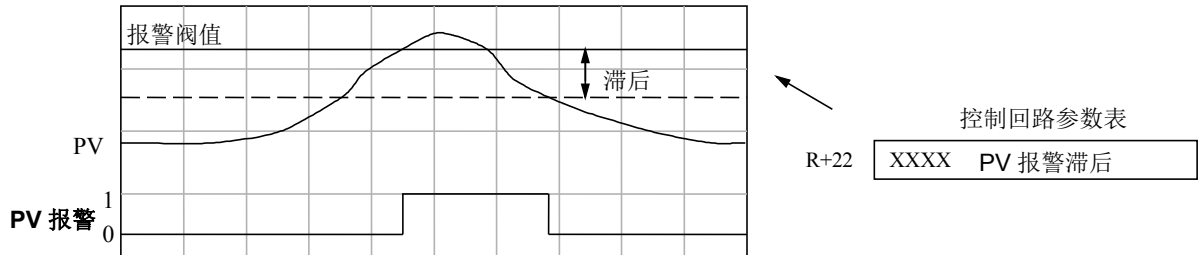
根据计算结果，在控制回路参数表的R+21，将变化率设置为“5”。通过PID模式1设置字R+00的Bit14设置启用/禁用PV变化率报警，且此报警独立于其他报警。

报警滞后功能(下面要讨论)不影响变化率报警。

四、PV 报警滞后

PV绝对值报警和PV偏差报警使用阈值进行编程。如果绝对值或偏差超过阈值，将产生报警。在实际运行中，PV信号带有干扰，将导致CPU的PV值波动。当PV值接近报警阈值，由于干扰引起PV值波动，报警将会持续不断。这可以通过使用PV报警滞后功能来解决。

PV报警滞后量可以设置在1~4095(BCD)的范围内。当使用PV偏差报警时，设置的滞后量必须低于设置的偏差量。下图显示了当PV值超过阈值并返回到阈值之下时，报警滞后如何起作用。



滞后范围是从超过阈值到返回到安全区的范围。这样，当PV值超过设置的报警阈值，将立即报警。PV值低于报警阈值，不会停止报警，直到PV值降至滞后值以下，才会停止报警。

五、报警设置出错

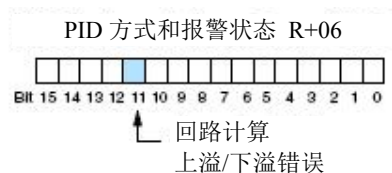
为使PV报警阈值有效，必须按照一定的数学关系，如下所示。如果不符合，如右图所示，报警设置错误位将被置位。

- PV绝对值报警要求：
低-低<低<高<高-高
- PV偏离量报警要求：
黄色<红色



六、回路计算上溢/下溢错误

当输出超出上限或下限，没有达到设定值时，就会发生错误。一个典型的例子是，当阀门卡住时，输出到极限了，但是PV值还没有达到设定值。



第十三节 上升/保持发生器

一、简介

在讨论基本PID回路操作时，根据不同的运行模式与编程方式，有各种方式生成PID控制设定值。下图中，上升/保持发生器是产生SP的一种方法。为了安全，在任何情况下，只有一个数据来源将SP值写入R+02。

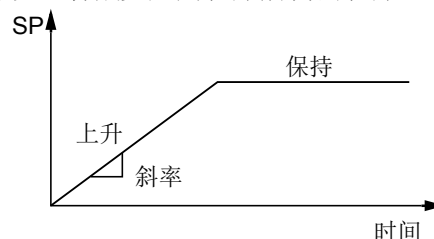
设定值来源：

操作员输入
上升/保持发生器
梯形图程序
其他回路输出(超级)



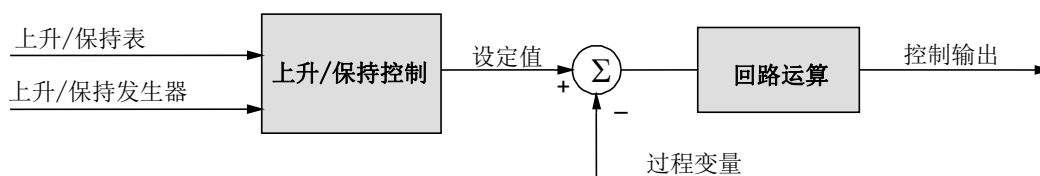
如果过程中的SP值很少变化，或者可以允许阶跃变化，则不需要使用上升/保持发生器。但是，有些过程必须精确控制SP值，使用上升/保持发生器可以显著减少应用程序所需的程序量。

在过程控制行业中，“上升”和“保持”有特殊的含义，温度控制应用中产生所需的设定值(SP)。如右图所示，在上升部分SP值增加，在保持部分恒定。



复杂的SP设置可以通过使用一系列上升/保持数据来完成。上升段设定每秒SP变化率，保持段以分钟为单位。

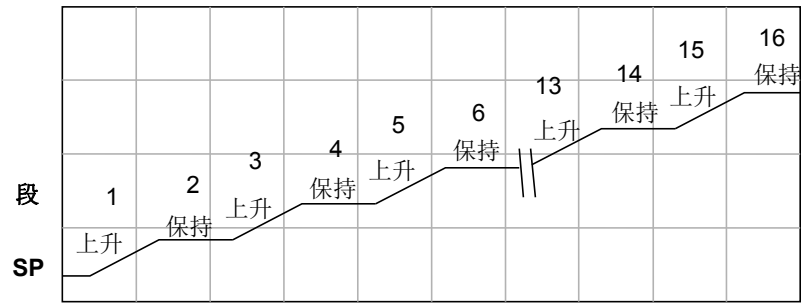
如下所示，上升/保持控制用于创建SP值，有两个输入部分。上升/保持表定义了上升/保持动作，必须先进行设置。在PID运算时，回路从该表中读取设置值。上升/保持控制由控制回路参数表特殊位来表示，实时控制上升/保持发生器的功能。可以通过梯形图程序监控上升/保持曲线的状态（当前斜率/区分编号）。



常见上升/保持运行的特点如下所示：

- 每个PID回路都有各自的上升/保持发生器(可任选)。
- 最大可以设定8个上升/保持(16段)。
- 当PLC处于RUN模式时，可以随时运行上升/保持发生器，与控制模式无关(手动或自动)。
- 上升/保持实时控制包括启动、锁定、继续和缓行工作。
- 上升/保持控制监控上升/保持曲线、保持段偏离(SP-PV)和当前上升/保持步号。

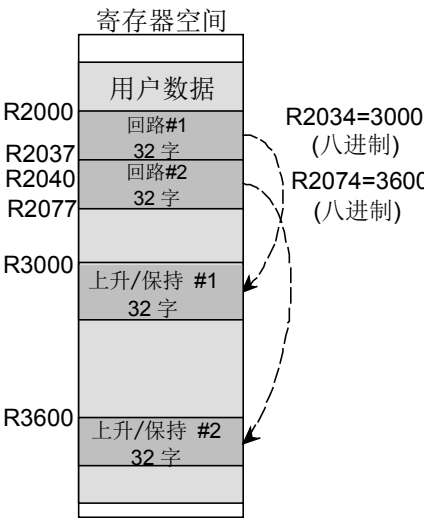
下图显示SP曲线由上升/保持段构成，共分16段。每段斜坡部分可以增加也可以减少。上升/保持发生器根据每段上升终点的相对值，自动是增加还是减少SP。这些值从上升/保持表读取。



二、上升/保持表

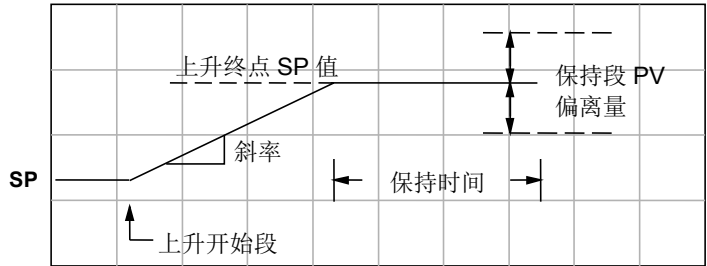
每个PID回路都可选择自己的上升/保持表，定义了上升/保持特性的参数。对于每个回路，控制参数由32个字存储器组成，占用一个连续存储区。用于PID控制的上升/保持表作为可选项，与控制回路参数表分开存储。在控制回路参数表的R+34中，存放了上升/保持表起始地址的指针。

在右边的示例中，PID回路#1和#2的控制回路参数表各自占用32个连续字的存储区。每个PID回路使用指针，在寄存器的其他地方存放各自的上升/保持表。当然，只要这些表不重叠，所有的表都可以存放在一个连续存储区。



上升/保持表的参数由用户定义，最方便的方法是使用KPP，有生成上升/保持表专用的编辑器。需要四个参数来定义一对上升和保持段，如下所示。

- 上升段终点SP值：规定上升终点的目标SP值。该值的数据类型必须与SP的数据类型相同。它可能大于或小于起始的SP值，所以斜率可能是上升或下降的(不需要知道上升段#1的起始SP值)。
- 上升段斜率：指定每秒SP变化率，它是BCD数，范围00.00-99.99(使用隐含的十进制小数点)。
- 保持段持续时间:指定保持段的时间，以分为单位，范围000.1-999.9分，BCD码(隐含十进制小数点)。
- 保持段PV偏离量:(可选项)指定了保持期间内PV偏离SP值的偏离量，PV偏离报警状态位由上升/保持发生器产生。



上升/保持表		
R+00	XXXX	上升段终点 SP 值
R+01	XXXX	上升段斜率
R+02	XXXX	保持段持续时间
R+03	XXXX	保持段 PV 偏离量

当前一个保持段结束后，下一个上升段就开始了。如果上升是第一段，则启用上升/保持发生器，并将当前SP设置为起始SP值。

地址偏移	步	说明	地址偏移	步	说明
+00	1	上升段终点SP值	+20	9	上升段终点SP值
+01	1	上升段斜率	+21	9	上升段斜率
+02	2	保持段持续时间	+22	10	保持段持续时间
+03	2	保持段PV偏离量	+23	10	保持段PV偏离量
+04	3	上升段终点SP值	+24	11	上升段终点SP值
+05	3	上升段斜率	+25	11	上升段斜率
+06	4	保持段持续时间	+26	12	保持段持续时间
+07	4	保持段PV偏离量	+27	12	保持段PV偏离量
+10	5	上升段终点SP值	+30	13	上升段终点SP值
+11	5	上升段斜率	+31	13	上升段斜率
+12	6	保持段持续时间	+32	14	保持段持续时间
+13	6	保持段PV偏离量	+33	14	保持段PV偏离量
+14	7	上升段终点SP值	+34	15	上升段终点SP值
+15	7	上升段斜率	+35	15	上升段斜率
+16	8	保持段持续时间	+36	16	保持段持续时间
+17	8	保持段PV偏离量	+37	16	保持段PV偏离量

许多应用不需要全部16段上升/保持，不使用的段在表中全部置零。当发现上升斜率为零后，上升/保持发生器停止动作。



注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

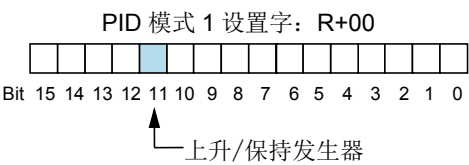
三、上升/保持表的标志

下表列出了上升/保持标志字(控制回路参数表中R+33)的各个Bit的定义。

Bit	上升/保持标志位的说明	读/写	Bit=0	Bit=1
0	启动上升/保持控制	写	—	0→1: 启动
1	锁定上升/保持控制	写	—	0→1: 锁定
2	继续上升/保持控制	写	—	0→1: 继续
3	点动上升/保持控制	写	—	0→1: 点动
4	上升/保持控制完成	读	—	结束
5	上升/保持PV输入偏离量错误	读	Off	On
6	上升/保持控制锁定状态	读	Off	On
7	保留	读	—	—
8-15	上升/保持控制的当前步	读	按字节编码(十六进制)	

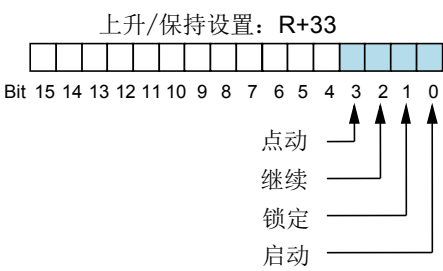
四、上升/保持程序控制有效

通过对PID模式1设置字R+00的Bit11的设定，如右图所示，可以使上升/保持发生器有效。只有上表所示的R+33的Bit0=1，上升/保持控制才能启动。



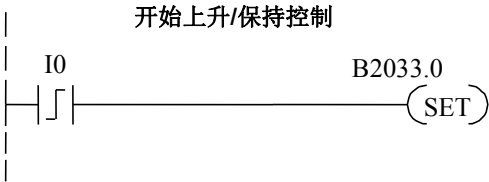
五、上升/保持控制

控制回路参数表上升/保持设置字的Bit0~3，用于设置上升/保持发生器的四种主要控制。在KPP中，这些位直接由上升/保持设置对话框控制。但是，在程序执行期间，必须使用梯形图程序来控制这些位，建议使用位操作指令。



在梯形图程序中，必须通过使用适当的指令，将控制位设置为“1”。当PID控制器读取上升/保持值时，这些设置位自动恢复为“0”。因此，当CPU处于RUN模式时，不需要重置这些位。

如右图所示，外部开关I0导通，使用其上升沿触点，用边沿设置适当的控制位，启动上升/保持曲线。这里我们使用位设置指令。



- 上升/保持控制位通常为零，梯形图程序一次只能设置一个控制位。
- 启动：一次0→1跳变，可启动上升/保持控制。CPU必须处于运行模式，PID回路可以处于手动或自动模式。如果控制过程没有应用锁定或缓行命令中断，上升/保持就能正常结束。
 - 锁定：一次0→1跳变，将停止上升/保持控制，保持当前状态，SP值将被冻结。
 - 继续：如果上升/保持控制处于锁定状态，一次0→1跳变，将使上升/保持发生器继续工作，SP值将使用当前的值。
 - 点动：一次0→1跳变，将使上升/保持发生器停止当前段执行，转到下一段。



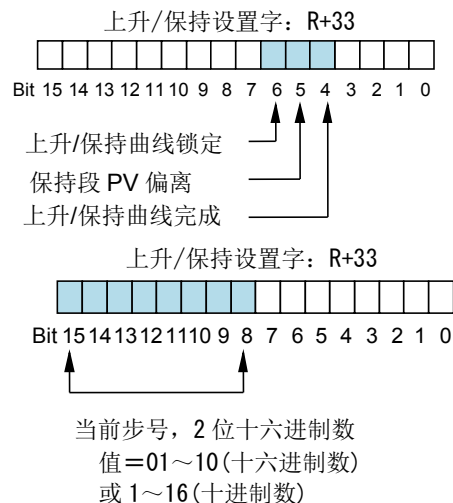
注意：DirectSOFT4、DirectSOFT5不支持SJ-Ether。
确保使用KPP Ver1.6.1.2或更高版本。

六、上升/保持程序控制的监控

可以用上升/保持设置字R+33中的其它位，来监视上升/保持控制状态，如右图所示。

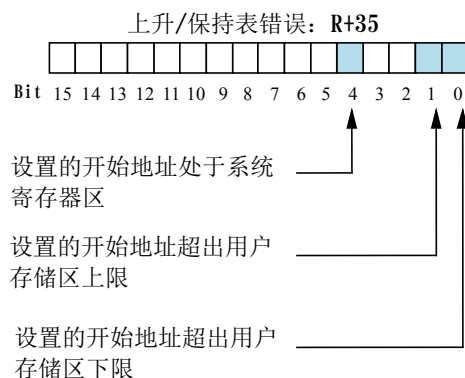
- 当最后一步完成后，上升/保持曲线完成位=1。
- 当误差（SP-PV）超过规定的偏离量时，保持段PV偏离=1。
- 当处于锁定状态时，上升/保持曲线锁定=1。

当前步号保存在上升/保持设置字R+33的高8位。这些位表示了2位十六进制数,范围1~10。梯形图程序可监视这些位,使程序的其它部分与上升/保持控制过程同步。把这个字装入累加器中并且右移8位,就能得到步号。



七、上升/保持编程错误

上升/保持表的起始地址必须是有效地址。如果设定的地址超出用户存储器范围，启动上升/保持发生器时，则如右图所示，其中的一位将置位。建议使用KPP来配置上升/保持表，KPP将自动检查地址范围。



八、测试上升/保持动作

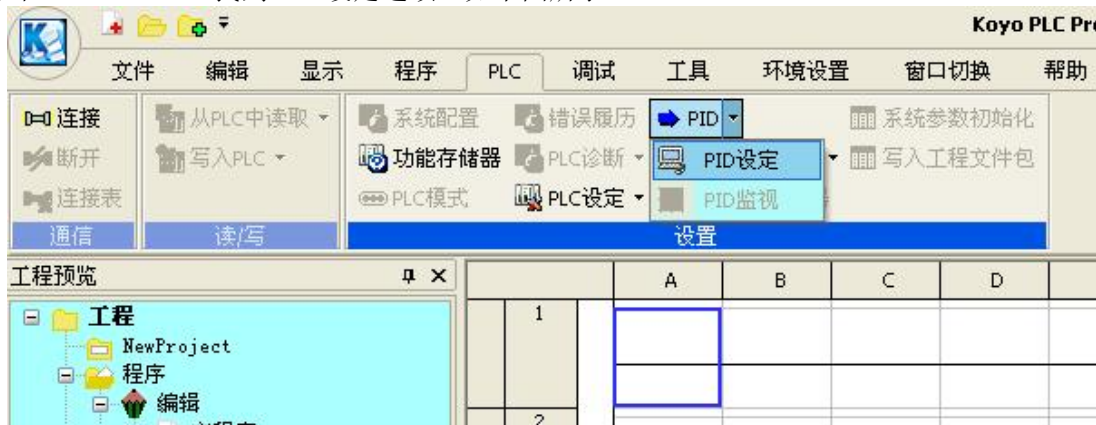
在使用上升/保持曲线控制过程之前，要先进行测试。即使PID回路处于手动模式，上升/保持发生器也会运行，因此测试可以轻松完成。通过使用KPP的PID视图，可以将曲线以图形方式显示。使趋势显示的时间足够缓慢，以便所有波形/保持段对可以显示在波形窗口中。

第十四节 利用KPPsoft软件设置PID参数

一、打开 PID 设定窗口

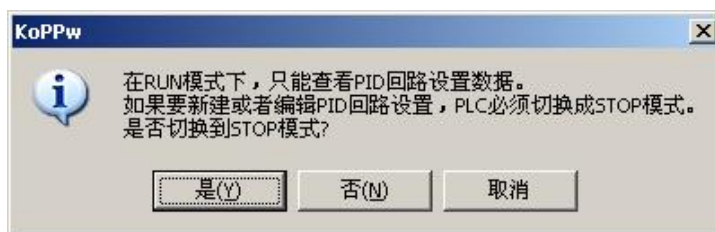
您当然可以用梯形图程序来设置 PID 回路的控制参数，但最方便的设置方法还是用 KPP 软件自带的 PID 设置工具。使用时，首先利用一个对话框设定好使用的回路数和参数表开始地址，然后用 PID 趋势监视画面对每个 PID 回路进行设置调整。设定和调整情报被自动放入对应的参数表中。

点击“PLC→PID”，找到 PID 设定选项，如下图所示：



如果这时 SJ-Ether 处于用户程序运行模式下，则 KPP 会弹出提示窗口如右所示。

选择[是]，使 SJ-Ether 切换到用户程序停止运行模式，并进入下一步。



如果连接的 PLC 之前没有进行过 PID 功能的设置，则会跳出 PID 数据存放区域的设置框：

“寄存器开始号”是指 PLC 中存放 PID 功能设定数据的寄存器的起始地址，“回路数”则是预设 PID 通道的个数 (SJ-Ether 系列 PLC 最多可设定 16 回路的 PID)，每个回路占用 32 个字节的寄存器空间。完成设定后，点击“写入完成”按钮保存设置。KPPSoft 编程工具软件自动弹出“PID 设定”窗口如下页。

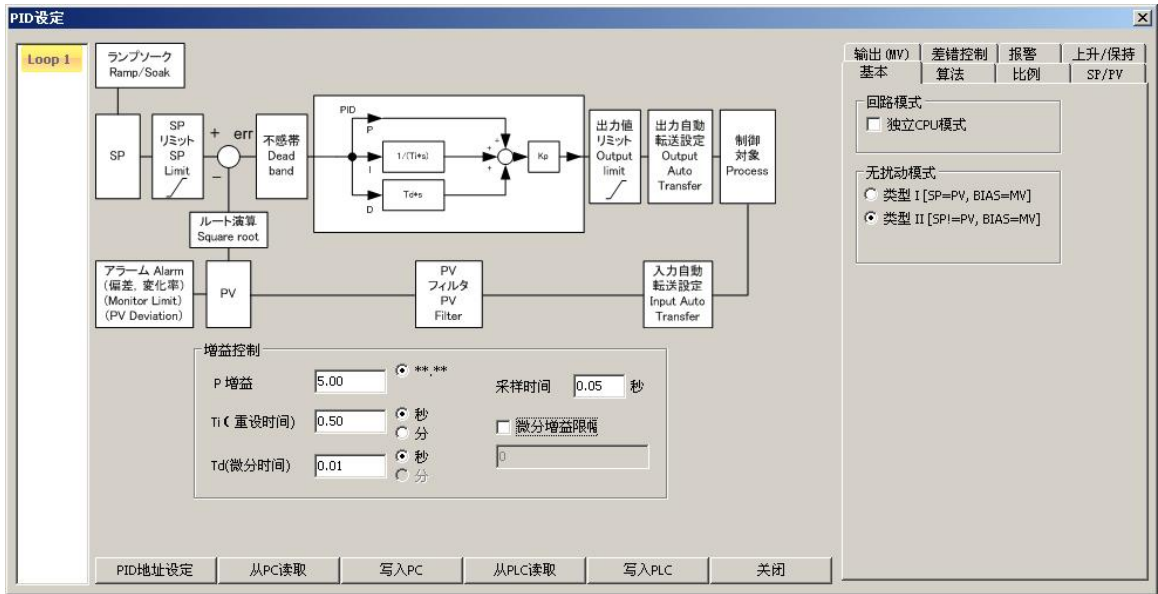
PID 功能设置窗口中有 1 个勾选项：

[从 PLC 再次读取 PID 数据]：指示在打开“PID 设定”窗口时，从 PLC 读出相关的参数数据。



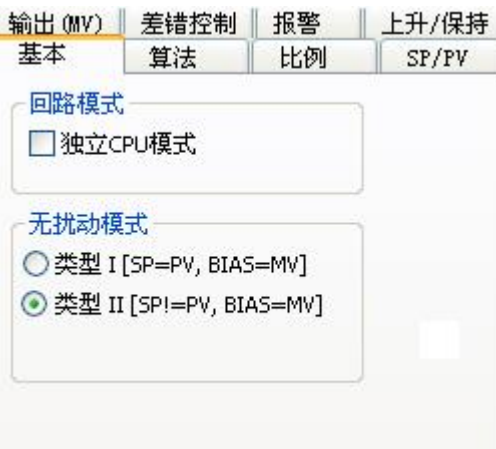
二、设置参数

“PID 设定”窗口由左右 2 部分组成，左边部分是针对每个回路的增益控制参数的设定以及使用的算法的图形表示，右边是针对每个 PID 回路的参数设置页，包括基本、算法、比例、SP/PV、输出(MV)、差错控制、报警、上升/保持等 8 个页。



下面介绍 8 页 PID 参数设置页的设置。

“基本”页主要是供用户进行回路模式与无扰动切换模式的选择。



回路模式：指定 PID 回路的运行是与 SJ-Ether 的运行模式相关联，还是独立于 SJ-Ether 运行模式。

无扰动模式：选择从手动 PID 运行模式到自动 PID 运行模式时的无扰动切换类型。

“算法”页则是用于对 PID 控制的方式和数据算法以及串级功能进行设定。

输出 (MV)

基本

差错控制

算法

报警

比例

上升/保持

SP/PV

算法

☒位置型

☐速度型

控制方向

☒正向控制

☐反向控制

控制类型

☒常规

☐串级

算法：选择 PID 控制回路的算法类型。
控制方向：选择 PID 控制回路的控制方向。
控制类型：选择 PID 控制的类型。

“比例”页设定输入输出数据格式。

输出 (MV)

基本

差错控制

算法

报警

比例

上升/保持

SP/PV

输入/输出数据格式

☒通用格式

☐单独格式

通用数据格式

12位数据格式

SP/PV正负

☒正

☐正负

输入/输出数据格式：指定输入/输出数据的格式，包括数据位数以及数据值的正负情况。
通用格式：输入/输出指定相同的数据格式。
单独格式：输入/输出分别指定各自的数据格式。
通用数据格式：指定通用格式下的输入/输出数据格式；
SP/PV 数据格式：选择是单极性还是双极性。
输出 (MV) 数据格式：选择“单独格式”下输出的数据格式。

“SP/PV” 页用于设定 SP（设定值）数值范围、PV（过程变量）采集方式。

输出 (MV)

基本

算法

比例

上升/保持

SP/PV

☒ SP值限制

SP上限: 4095 4095

SP下限: 0 0

应用

☐ PV线性/平方根选择

SP 值限制: 设置 SP 值的最大最小值，防止非法 SP 值的输入。

PV 线性/平方根选择: 选择如何从过程变量得到 PV 值的方法，可选择常规数据（线性）或开平方后的数据。（勾选表示选择平方根数据）

“输出 (MV)” 页用于设定 PID 控制的数据输出范围。

基本

输出 (MV)

算法

比例

SP/PV

差错控制

报警

上升/保持

输出(MV)限制

MV上限: 4095 4095

MV下限: 0 0

应用

输出 (MV) 限制: 设置输出值的最大最小值，防止超范围数据的输出。

“差错控制” 页用于设定平方误差、误差死区等功能参数。

基本

输出 (MV)

算法

比例

SP/PV

差错控制

报警

上升/保持

误差

☐ 平方误差

☐ 误差死区

死区大小 0

☐ 偏差冻结

平方误差: 勾选选择把误差的平方作为误差项使用；不勾选误差项直接使用线性误差。

误差死区: 设置使用误差死区，以及死区的数值大小。

偏差冻结: 选择使用积分分离功能，以使得在控制输出超出限制值时，使偏差值冻结，积分起作用。

“报警”页中可进行 PID 控制中所需的报警临界值等数据的设置。

基本

算法

比例

SP/PV

输出(MV)

差错控制

报警

上升/保持

☒PV值报警

高-高: 1500

高: 1000

低: 500

低-低: 100

清除

☒PV偏差报警

红: 1000

黄: 100

清除

☒PV值变化率报警

0

报警滞后

值: 0

PV 值报警：设置是否使用 PV 绝对值报警，以及其高-高、高、低、低-低报警值的设置。

PV 偏差报警：设置是否使用 PV 值与 SP 值间的偏差值范围报警，以及其报警范围的设置。

PV 变化率报警：设置是否使用 PV 变化率报警，以及设置其报警变化率大小。

报警滞后：设置 PV 报警的滞后量。

“上升/保持”页用于在使用 PID 的上升/保持功能时，设定 PID 运行的上升/保持相关参数，点击“位置”后的“...”按钮可以设定对应于 PLC 数据存放区域，使用“向导”按钮可进入设定向导页面。

在这儿可设置使用上升/保持（RAMP/SOAK）算法控制 SP 值变化时的上升/保持算法参数表参数。

基本

算法

比例

SP/PV

输出(MV)

差错控制

报警

上升/保持

位置:

???? - ????

有效

向导

步

0

0

0

0

0

0

0

0

0

0

0

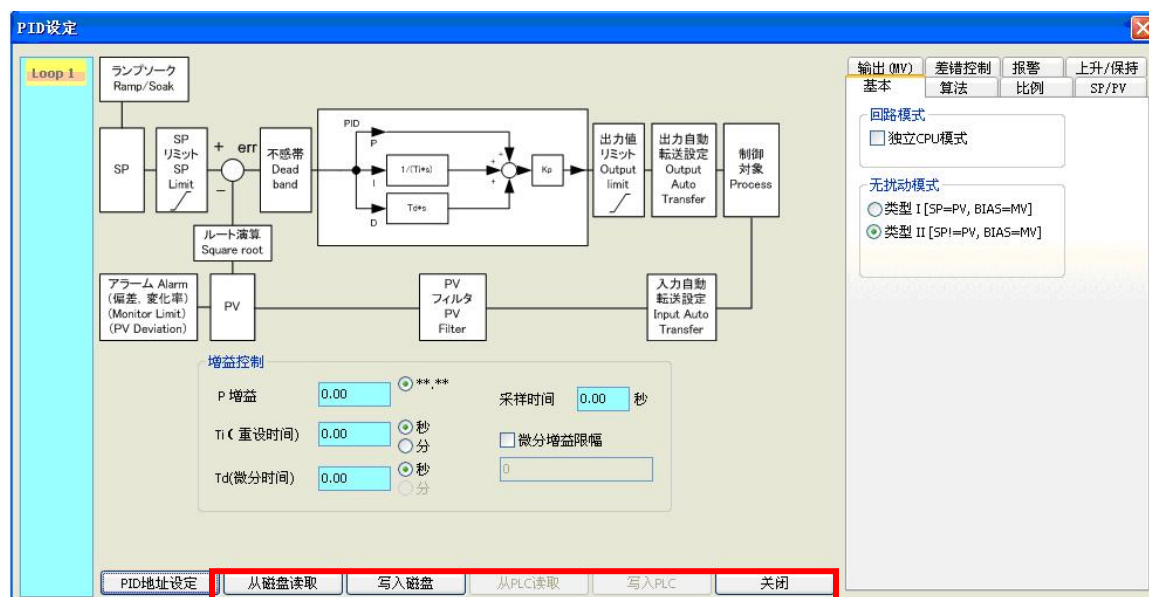
0

0

0

	上升	保持		
步	SP	斜率	时间(分)	偏差
1 & 2	0	0.00	0.0	0
3 & 4	0	0.00	0.0	0
5 & 6	0	0.00	0.0	0
7 & 8	0	0.00	0.0	0
9 & 10	0	0.00	0.0	0
11 & 12	0	0.00	0.0	0
13 & 14	0	0.00	0.0	0
15 & 16	0	0.00	0.0	0

以上参数全部设定完成后，点击下图红色方框中的“写入 PLC”按钮，将数据保存到 PLC 中。还可以点击“写入磁盘”按钮，把设置参数保存到计算机磁盘。注意：必须在连接上 NK1 系列 PLC 后，下图中的[写入 PLC]按钮才可选。



点击“关闭”按钮关闭 PID 设定窗口。注意点击“关闭”按钮时，如还未保存有关设置内容，则 KPP 会弹出提醒保存有关内容的提示窗，点击“确定”即可。

注意：如果用以上的方法来设置各 PID 参数，为了长久保存这些设置好的 PID 参数，分配给各 PID 参数用的寄存器区必须设置成停电保持。例子中 R2000~R2037 这 32 个寄存器必须设置为停电保持寄存器。

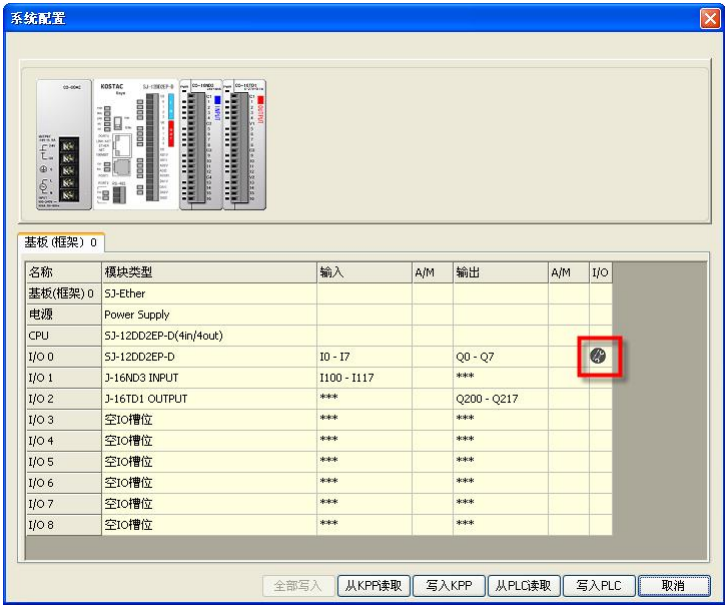
三、PID 设置及编程举例

通过上面的步骤设置好 PID 参数后，为了让 PID 回路功能启用，还需要编制一段 PLC 用户程序，以设置 PID 回路的 SP 值、启动 PID 自动运行模式、以及取得 PID 回路 PV 值和把控制输出值送到模拟量输出通道中去等等。CPU 自带模拟量通道可以通过设置对话框设置，也可以用梯形图设置。

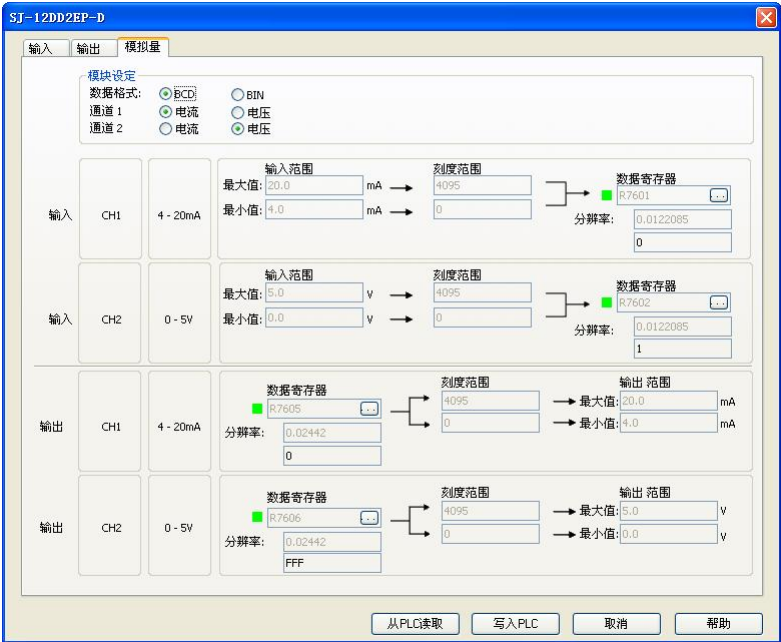
下面给出一个 PID 设置及编程示例：

在本例中，SJ-Ether 系统由本体单元 SJ-12DD2EP-D(带 4 点开关量输入、4 点开关量输出、2 通道模拟量输入、2 通道模拟量输出)、16 点输入模块 J-16ND3 和 16 点输出模块 J-16TD1 组成，SJ-Ether 的 PID 功能使用一个 PID 回路，PID 回路参数起始地址为 R2000，使用 1 号输入通道（4~20mA）与 2 号输出通道（0~5V）作为 PID 回路的 PV 值来源和控制输出。

其系统组成图如下。



如果通过对话框设置模拟量，可以点击上图中红框标出标志，弹出如下对话框。注意：输入通道 1 与输出通道 1 必须相同设置，输入通道 2 与输出通道 2 也必须相同设置。



模拟量通道也可以通过梯形图程序设置。

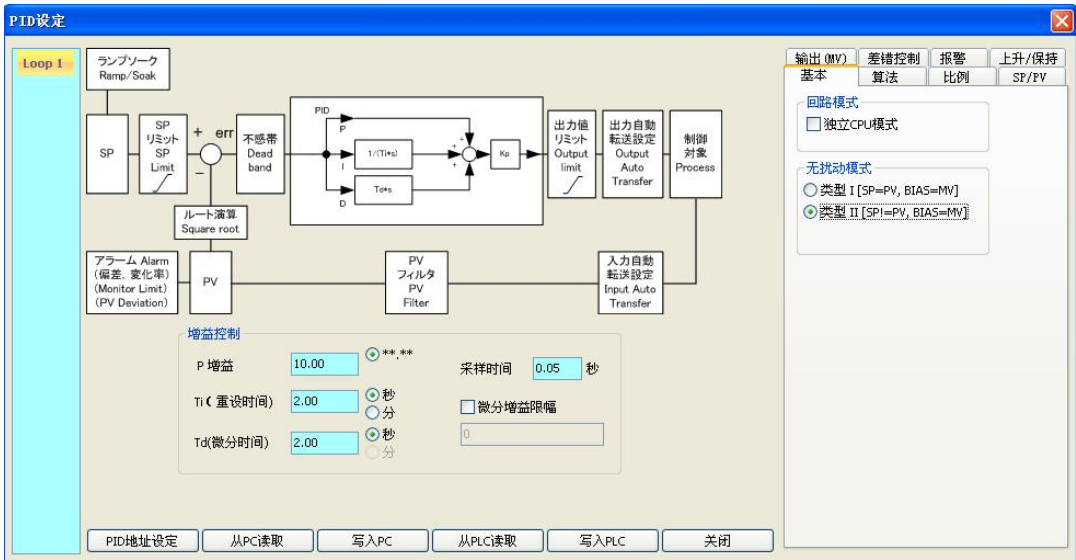
PID 参数设置按以下图示进行。

首先设置使用 PID 回路数及 PID 参数寄存器区起始地址。



按[写入完成]按钮，弹出[PID 设定] 窗口，进行各种 PID 参数设置如下。

先按下图左边部分，设置好每个回路的增益控制参数。



后设置 PID 回路的 8 页 PID 基本参数如下：

基本

输出 (MV)	差错控制	报警	上升/保持
基本	算法	比例	SP/PV

回路模式

☐ 独立CPU模式

无扰动模式

☐ 类型 I [SP=Pv, BIAS=MV]

☒ 类型 II [SP!=Pv, BIAS=MV]

设置为：不使用独立回路；无扰动切换模式为“类型 II”。

算法

输出 (MV)	差错控制	报警	上升/保持
基本	算法	比例	SP/PV

算法

☒ 位置型

☐ 速度型

控制方向

☒ 正向控制

☐ 反向控制

控制类型

☒ 常规

☐ 串级

选用位置型算法，控制方向为正向控制，控制类型为常规。

比例

输出 (MV)	差错控制	报警	上升/保持
基本	算法	比例	SP/PV

输入/输出数据格式

☒ 通用格式

☐ 单独格式

通用数据格式

12位数据格式

SP/PV正负

☒ 正

☐ 正负

使用通用的 12 位数据格式，并仅指定 SP/PV 值为正数据。

SP/PV

输出 (MV)	差错控制	报警	上升/保持
基本	算法	比例	SP/PV

☒ SP值限制

SP上限: 4095 4095

SP下限: 0 0 应用

☐ PV线性/平方根选择

设定 SP 值上限为 4095，下限为 0；采用常规过程变量数据作为 PV 值。

输出

基本

算法

比例

SP/PV

输出 (MV)

差错控制

报警

上升/保持

输出(MV)限制

MV上限:40954095

MV下限:00

应用

设定输出（MV）值上限为 4095，下限为 0。

差错控制

基本

算法

比例

SP/PV

输出 (MV)

差错控制

报警

上升/保持

误差

☐平方误差

☐误差死区

死区大小0

☐偏差冻结

差错控制项都不勾选。

报警

基本

算法

比例

SP/PV

输出 (MV)

差错控制

报警

上升/保持

☒PV值报警

高-高:1500

高:1000

低:500

低-低:100

清除

☒PV偏差报警

红:1000

黄:100

清除

☒PV值变化率报警

10

报警滞后

值:0

设置各 PV 值报警和 PV 偏差报警如上图；PV 变化率报警值为 10，报警滞后没有设置。

上升/保持

基本

算法

比例

SP/PV

输出 (MV)

差错控制

报警

上升/保持

位置:????-????

☐有效

向导

步

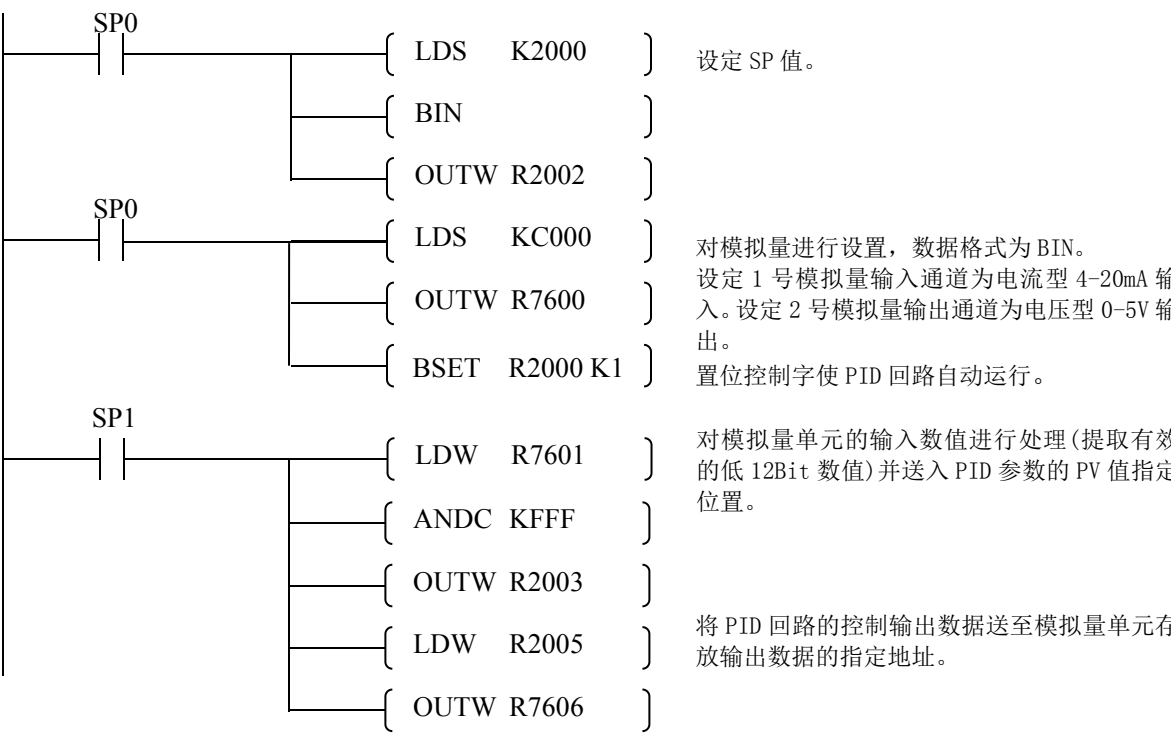
上升

保持

步	SP	斜率	时间(分)	偏差
1 & 2	0	0.00	0.0	0
3 & 4	0	0.00	0.0	0
5 & 6	0	0.00	0.0	0
7 & 8	0	0.00	0.0	0
9 & 10	0	0.00	0.0	0
11 & 12	0	0.00	0.0	0
13 & 14	0	0.00	0.0	0
15 & 16	0	0.00	0.0	0

没有使用上升/保持算法控制 SP 值变化功能。

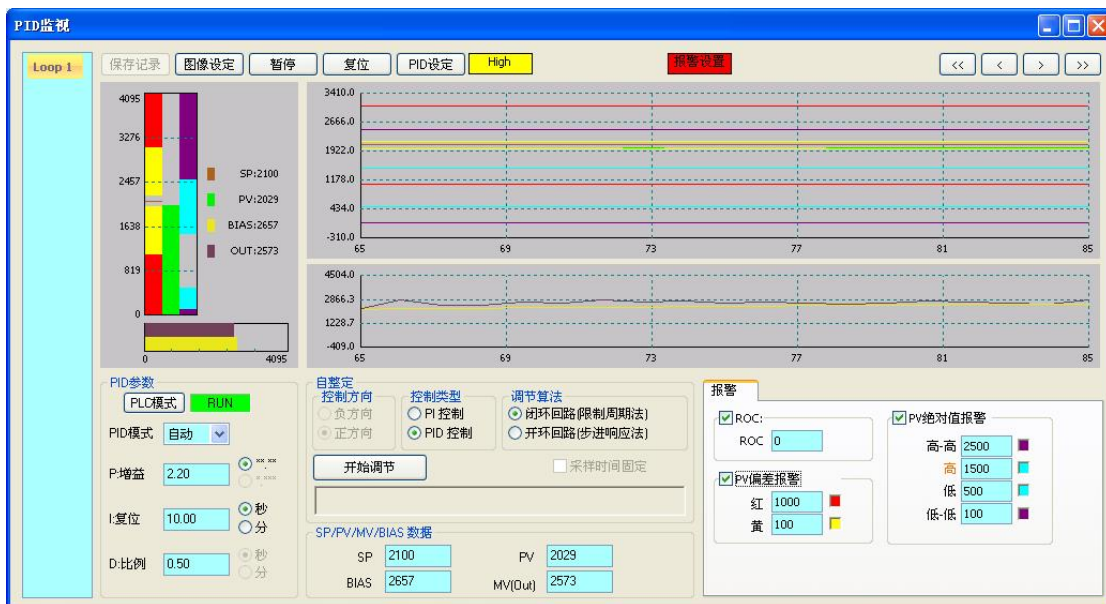
上面的 PID 参数设置完成后，为了让 PID 回路功能启用，还需要编制一段 PLC 用户程序，以设置一些基本的模拟量单元工作参数(此处用梯形图设置)、启动 PID 自动运行模式、以及取得 PID 回路 PV 值和把控制输出值送到模拟量输出通道中去等。



注意：上面的例子中只设置了 1 路 PID 回路，所以只有对该 1 路 PID 回路参数进行设置的说明，当然，也只有针对该 1 路 PID 回路的程序例子。当应用中设置有多路 PID 回路时，需要对每路 PID 回路进行如上的设定并编制必要的用户程序。

四、PID 运行监视

PLC 的 PID 控制运行后，可利用 KPP 的 PID 监控窗对 PID 运行状态进行监控。点击“PID→PID 监视”。



利用此视图，可以查看 PID 的控制曲线，也可以对一些控制参数进行在线编辑调整，以优化控制算法。在这儿，你也可以启动 SJ-Ether 系列 PLC 的 PID 自整定功能，以自动得到 PID 算法的各回路增益控制参数。

第十五节 故障检修提示

Q. 回路不进入自动方式。

A. 检查下列可能的原因：

- PLC在编程模式。回路要运行，必须使PLC在RUN模式。
- 存在PV报警，或PV报警程序发生了错误。
- 回路为串级回路的主环，副环不在串级模式下。

Q. 当回路处于自动模式，控制输出始终为零。

A. 检查下列可能的原因：

- 控制回路参数表R+31中的控制输出上限是零。
- 所有的整定参数都为0。
- 回路达到饱和，因为误差不可能是零并且其正负值也会改变。

Q. 控制输出值不是零，但也不正确。

A. 检查下列可能的原因：

- 增益值是否正确？PID控制回路参数表中增益是BCD数，而SP和PV是二进制数。在KPP中，可以将SP、PV及控制输出的二进制数转换为BCD数显式。

Q. 当激活启动位时，上升/保持发生器不工作。

A. 检查下列可能的原因：

- 上升/保持使能位是否为“0”？检查控制回路参数表R+00的第11位，该位必须被置为“1”。
- 在上升/保持控制中，锁定位或其它位是否为“1”？
- 起始的SP值和第一个上升段结束时的SP值相同。第一个上升段没有斜率而且没有持续周期。上升/保持发生器很快移到保持段，造成第一个上升段不工作的假象。
- 回路处于串级模式，试图读取远程的SP。
- 在控制回路参数表R+27中的SP上限值太低。
- 是否SP值没有写入控制回路参数表R+02？手动模式下启动上升/保持发生器进行确认。

Q. 即使模拟量模块收到了PV信号，在表中的PV值仍是不变。

A. 梯形图程序必须成功地从模块中得到模拟量值并把它写到控制回路参数表R+03的地址中。确认模拟量模块工作正常，而且梯形图正在工作。

Q. 微分增益似乎对输出没有影响。

A. 可能达到了微分限幅值（见有关微分增益限制值一段）。

Q. 回路设定值自己变化。

A. 检查下列可能的原因：

- 上升/保持发生器起作用，并且正在产生设定值。
- 如果回路从手动-自动方式转换，那么回路自动设置为SP=PV（无扰动转换特性）。
- 检查梯形图程序，确认它没有对SP地址（控制回路参数表中的R+02）进行写操作。进行这些工作最快的办法就是在程序开头临时放一个结束线圈，然后进入PLC运行方式。

Q. 用KPP软件输入的SP和PV值能正常工作，但当梯形图程写数据时这些值没有工作。

A. KPP软件的PID视图，可用十进制输入SP、PV和偏差值并且以十进制显示出来。例如，当数据类型是单极性12位时，数值范围从0~4095。但是，控制回路参数表实际上需要十六进制的数，KPP软件会自动进行转换。在表中的数值的范围从0~FFF(12位单极性类型)。

Q. 无论使用什么方法，回路似乎不稳定而且无法调整。

A. 检查下列可能的原因：

- 回路采样时间被设得太长。参见本章前面的有关选择回路刷新时间的段落。
- 增益太高。把微分增益减小到零，必要的话依次减小积分增益、比例增益。
- 此过程有很大的响应延迟。这意味着PV对控制输出的变化反应缓慢，可能是由于执行

机构和PV传感器之间的“距离”太长，或执行机构传输能量到过程的能力太弱。

- 可能有超过回路调整能力的过程干扰。当SP不变化时，保证PV相对稳定。

附录1 PID回路术语汇总

自动方式	回路的一种操作方式，在此方式下，回路进行 PID 运算并刷新回路的控制输出。
偏差冻结	当控制输出超出范围时，将偏差值保留下来，抑制积分作用，其好处在于使回路调节恢复得更快。
偏差项	PID 公式的位置方式，它是积分器和初始控制输出值的和。
无扰动切换	改变回路工作方式的一种方法，以避免控制输出的突变。通过人工地使 $SP=PV$ ，或在方式改变的那一刻使偏差=控制输出来避免突变。
串级回路	串级回路把另一个回路的输出作为其设定值。串级回路有主/副关系，并且共同工作并最终控制一个 PV。
串级方式	回路的一种工作方式，在此方式下，它使用另一个回路的输出作为其 SP。
连续控制	过程控制是通过传送平滑的（模拟量）信号作为控制输出来完成的。
正作用回路	PV 增加，控制输出也增加的回路，换句话说，过程有正增益。
误差	SP 与 PV 之间的差值，误差=SP-PV。
误差死区	当误差很小时，回路对误差灵敏度的选择特性，可以规定死区的大小。
误差平方	误差乘以误差的可选特性，但是保留最初的运算符号，它减小了小误差的影响，而扩大了大误差的作用。
前馈	当设定值改变，或输入信号有干扰时，对回路的控制进行优化的一种方法。
控制输出	PID 算式的运算结果。
微分增益	决定了 PID 微分项的大小。
积分增益	决定了 PID 积分项的大小。
主环	在串级控制中，它就是产生串级回路设定值的回路。
手动方式	回路的一种工作方式，在此方式下，PID 运算被禁止。操作者必须直接写入控制输出值来控制回路。
副环	在串级控制中，副环就是从主环中接收其 SP 的副回路。
On/Off 控制	控制过程的一种简单的方法，将整个过程化为对 on/off 的响应，以得到相对平滑的 PV。一个简单的梯形程序就可以把 NK1 的连续回路输出转换成 on/off 控制。
PID 回路	一种闭环回路控制的数学方法，其中包含三个基本项：比例，积分和微分项的和，这三项都有其独立的增益值，可以用其中一项来优化（调整）回路以得到某一特殊的应用系统。
位置运算	计算出控制输出，使它响应 PV 对于 SP（误差项）的偏离量（位置）。
过程	把数据值施加到原材料上去的控制过程，过程控制尤指过程中引起的材料的化学变化。
过程变量（PV）	过程中材料的物理特性的一种定量的测量值，它会影响最终的产品质量因此监视和控制它是十分重要的。
比例增益	这个常量决定了当前误差的 PID 比例项的大小。
PV 绝对值报警	把 PV 值与报警阈值进行比较的一种可编程报警。
PV 偏离量报警	把 SP 与 PV 的差值与偏离极限值进行比较的一种可编程报警。
上升/保持控制过程	一系列的 SP 值称作一个控制过程，它在每次回路计算时实时产生，控制包括一系列成对的上升和保持段。
速率	也叫微分器，响应误差项的变化的速率。
间接设定值	在串级回路中当回路被设置成副环时，它读取其设定值的地址。
复位	也称作积分器，复位项就是把每次被采样到的误差与上一次的误差相加，保持一个偏差累加值。

积分饱和	当回路无法达到平衡时并且误差使积分（复位）累加过剩（超调）时,出现的一种情况。当原始回路故障被修复后，积分饱和将使恢复时间延长。
反作用回路	PV 增加控制输出减少的回路，换句话说，过程有负增益。
采样时间	PID 运算间隔时间，过程控制的 CPU 方式被称为采样控制器，因为它只周期地采集 SP 和 PV。
设定值（SP）	过程变量的理想值，在闭环回路工作时，设定值（SP）是回路控制器的输入值。
保持偏离量	保持偏离量是当上升/保持程序控制工作时，在上升/保持控制过程处于保持段内时测量的 SP 与 PV 间的偏差值。
步序响应	过程变量响应 SP 的步序变化（闭环工作时），或控制输出的步序变化（开环工作时）的工作情况。
切换	把回路从一种工作方式变成另一种工作方式（手动，自动或串级）。
速度算法	使 PV 等于 SP 的控制输出，表达成 PV 的变化速率（速度）。



捷太格特电子(无锡)有限公司

JTEKT ELECTRONICS (WUXI) CO.,LTD.

地址：江苏省无锡市滨湖区建筑西路 599 号 1 栋 21 层 邮编：214072
电话：0510-85167888 传真：0510-85161393
网址：<https://www.jtektele.com.cn>

JELWX-M3727B

2024 年 7 月