

如何使用 S7-200CPU 的 PID 控制

第一部分、PID 控制

S7-200 能够进行 PID 控制。S7-200 CPU 最多可以支持 8 个 PID 控制回路（8 个 PID 指令功能块）。

PID 是闭环控制系统的比例—积分—微分控制算法。

PID 控制器根据设定值（给定）与被控对象的实际值（反馈）的差值，按照 PID 算法计算出控制器的输出量，控制执行机构去影响被控对象的变化。

PID 控制是负反馈闭环控制，能够抑制系统闭环内的各种因素所引起的扰动，使反馈跟随给定变化。

根据具体项目的控制要求，在实际应用中有可能用到其中的一部分，比如常用的是 PI（比例—积分）控制，这时没有微分控制部分。

PID 算法在 S7-200 中的实现

PID 控制最初在模拟量控制系统中实现，随着离散控制理论的发展，PID 也在计算机化控制系统中实现。

为便于实现，S7-200 中的 PID 控制采用了迭代算法。详细的计算方法请参考《S7-200 系统手册》中 PID 指令部分的相关内容。

计算机化的 PID 控制算法有几个关键的参数 K_c （Gain，增益）， T_i （积分时间常数）， T_d （微分时间常数）， T_s （采样时间）。

在 S7-200 中 PID 功能是通过 PID 指令功能块实现。通过定时（按照采样时间）执行 PID 功能块，按照 PID 运算规律，根据当时的给定、反馈、比例—积分—微分数据，计算出控制量。

PID 功能块通过一个 PID 回路表交换数据，这个表是在 V 数据存储区中的开辟，长度为 36 字节。因此每个 PID 功能块在调用时需要指定两个要素：PID 控制回路号，以及控制回路表的起始地址（以 VB 表示）。

由于 PID 可以控制温度、压力等等许多对象，它们各自都是由工程量表示，因此有一种通用的数据表示方法才能被 PID 功能块识别。S7-200 中的 PID 功能使用占**调节范围**的百分比的方法抽象地表示被控对象的数值大小。在实际工程中，这个调节范围往往被认为与被控对象（反馈）的测量范围（量程）一致。

PID 功能块只接受 0.0 - 1.0 之间的实数（实际上就是百分比）作为反馈、给定与控制输出的有效数值，如果是直接使用 PID 功能块编程，必须保证数据在这个范围之内，否则会出错。其他如增益、采样时间、积分时间、微分时间都是实数。

因此，必须把外围实际的物理量与 PID 功能块需要的（或者输出的）数据之间进行转换。这就是所谓输入/输出的转换与标准化处理。《S7-200 系统手册》上有详细的介绍。

S7-200 的编程软件 Micro/WIN 提供了 PID 指令向导，以方便地完成这些转换/标准化处理。除此之外，PID 指令也同时会被自动调用。

调试 PID 控制器

PID 控制的效果就是看反馈（也就是控制对象）是否跟随设定值（给定），是否响应快速、稳定，是否能够抑制闭环中的各种扰动而回复稳定。

要衡量 PID 参数是否合适，必须能够连续观察反馈对于给定变化的响应曲线；而实际上 PID 的参数也是通过观察反馈波形而调试的。因此，没有能够观察反馈的连续变化波形曲线的有效手段，就谈不上调试 PID 参数。

观察反馈量的连续波形，可以使用带慢扫描记忆功能的示波器（如数字示波器），波形记录仪，或者在 PC 机上做的趋势曲线监控画面等。

新版编程软件 STEP 7 - Micro/WIN V4.0 内置了一个 PID 调试控制面板工具，具有图形化的给定、反馈、调节器输出波形显示，可以用于手动调试 PID 参数。对于没有“自整定 PID”功能的老版 CPU，也能实现 PID 手动调节。

PID 参数的取值，以及它们之间的配合，对 PID 控制是否稳定具有重要的意义。这些主要参数是：

• 采样时间：

计算机必须按照一定的时间间隔对反馈进行采样，才能进行 PID 控制的计算。采样时间就是对反馈进行采样的间隔。短于采样时间间隔的信号变化是不能测量到的。过短的采样时间没有必要，过长的采样间隔显然不能满足扰动变化比较快、或者速度响应要求高的场合。

编程时指定的 PID 控制器采样时间必须与实际的采样时间一致。S7-200 中 PID 的采样时间精度用定时中断来保证。

• 增益（Gain，放大系数，比例常数）

增益与偏差（给定与反馈的差值）的乘积作为控制器输出中的比例部分。过大的增益会造成反馈的振荡。

• 积分时间 (Integral Time)

偏差值恒定时，积分时间决定了控制器输出的变化速率。积分时间越短，偏差得到的修正越快。过短的积分时间有可能造成不稳定。

积分时间的长度相当于在阶跃给定下，增益为“1”的时候，输出的变化量与偏差值相等所需要的时间，也就是输出变化到二倍于初始阶跃偏差的时间。

如果将积分时间设为最大值，则相当于没有积分作用。

• 微分时间 (Derivative Time)

偏差值发生改变时，微分作用将增加一个尖峰到输出中，随着时间流逝减小。微分时间越长，输出的变化越大。微分使控制对扰动的敏感度增加，也就是偏差的变化率越大，微分控制作用越强。微分相当于对反馈变化趋势的预测性调整。

如果将微分时间设置为 0 就不起作用，控制器将作为 PI 调节器工作。

常问问题

1、对于某个具体的 PID 控制项目，是否可能事先得知比较合适的参数？有没有相关的经验数据？

虽然有理论上计算 PID 参数的方法，但由于闭环调节的影响因素很多而不能全部在数学上精确地描述，计算出的数值往往没有什么实际意义。因此，除了实际调试获得参数外，没有什么可用的经验参数值存在。甚至对于两套看似一样的系统，都可能通过实际调试得到完全不同的参数值。

2、PID 控制不稳定怎么办？如何调试 PID？

闭环系统的调试，首先应当做开环测试。所谓开环，就是在 PID 调节器不投入工作的时候，观察：

- 反馈通道的信号是否稳定
- 输出通道是否动作正常

可以试着给出一些比较保守的 PID 参数，比如放大倍数（增益）不要太大，可以小于 1，积分时间不要太短，以免引起振荡。在这个基础上，可以直接投入运行观察反馈的波形变化。给出一个阶跃给定，观察系统的响应是最好的方法。

如果反馈达到给定值之后，历经多次振荡才能稳定或者根本不稳定，应该考虑是否增益过大、积分时间过短；如果反馈迟迟不能跟随给定，上升速度很慢，应该考虑是否增益过小、积分时间过长……

总之，PID 参数的调试是一个综合的、互相影响的过程，实际调试过程中的多次尝试是非常重要的步骤，也是必须的。

S7-200 的新一代产品提供了自整定的 PID 细调功能。

3、没有采用积分控制时，为何反馈达不到给定？

这是必然的。因为积分控制的作用在于消除纯比例调节系统固有的“静差”。没有积分控制的比例控制系统中，没有偏差就没有输出量，没有输出就不能维持反馈值与给定值相等。所以永远不能做到没有偏差。

4、如何实现 PID 反作用调节？

参见 [PID 向导中的常问问题](#)。

5、S7-200 控制变频器，在变频器也有 PID 控制功能时，应当使用谁的 PID 功能？

可以根据具体情况使用。一般来说，如果需要控制的变量直接与变频器直接有关，比如变频水泵控制水压等，可以优先考虑使用变频器的 PID 功能。

6、《S7-200 系统手册》上的附录 H.14 “用 S7-200 实现 PID 控制”的例子，是否可以直接使用？

《S7-200 系统手册》中的附录 H 在英文原版中并不存在。

H.14 的 PID 例子是在第一代产品还不支持 PID 运算指令时的产物。现在用户可以使用 PID 指令块，或者 PID Wizard (PID 向导) 编辑 PID 控制程序。

PID Wizard – PID 向导

Micro/WIN 提供了 PID Wizard (PID 指令向导)，可以帮助用户方便地生成一个闭环控制过程的 PID 算法。此向导可以完成绝大多数 PID 运算的自动编程，用户只需在主程序中调用 PID 向导生成的子程序，就可以完成 PID 控制任务。

PID 向导既可以生成模拟量输出 PID 控制算法，也支持开关量输出；既支持连续自动调节，也支持手动参与控制。**建议用户使用此向导对 PID 编程，以避免不必要的错误。**如果用户不能确定中文编程界面的语义，我们建议用户使用英文版本的 Micro/WIN，以免对向导中相关概念发生误解。

建议用户使用较新的编程软件版本。在新版本中的 PID 向导获得了改善。

PID 向导编程步骤

在 Micro/WIN 中的命令菜单中选择 **Tools > Instruction Wizard**，然后在指令向导窗口中选择 PID 指令：

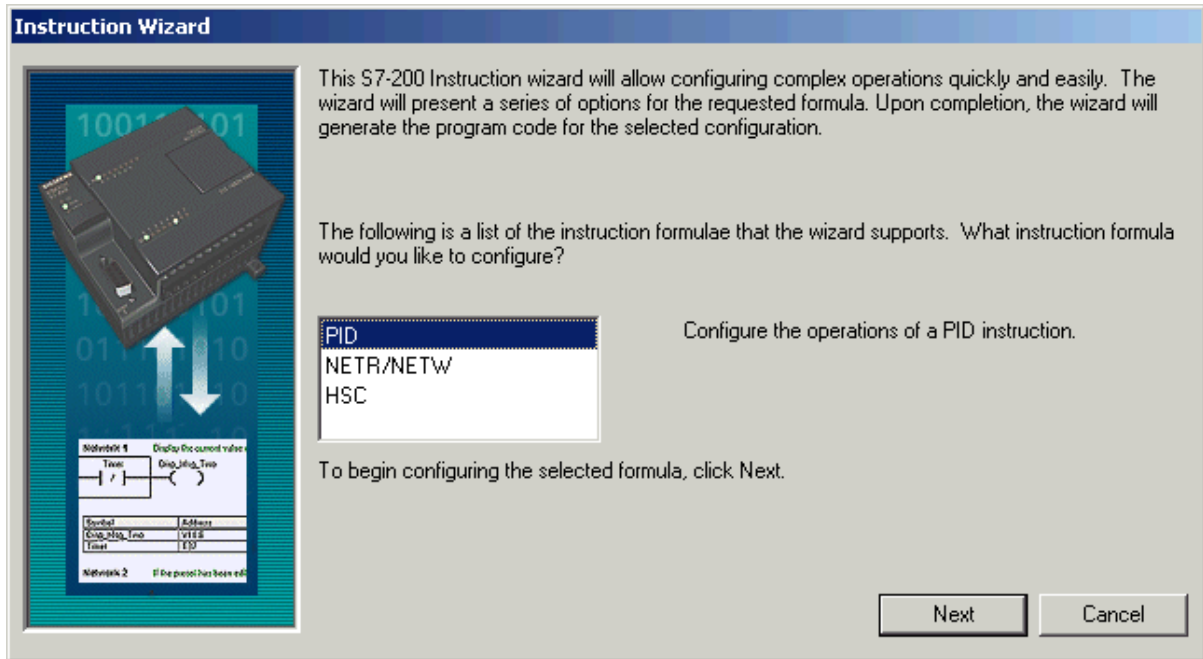


图 1. 选择 PID 向导

在使用向导时必须先对项目进行编译，在随后弹出的对话框中选择“**Yes**”，确认编译。如果已有的程序中存在错误，或者有没有编完的指令，编译不能通过。

如果你的项目中已经配置了一个 PID 回路，则向导会指出已经存在的 PID 回路，并让你选择是配置修改已有的回路，还是配置一个新的回路：

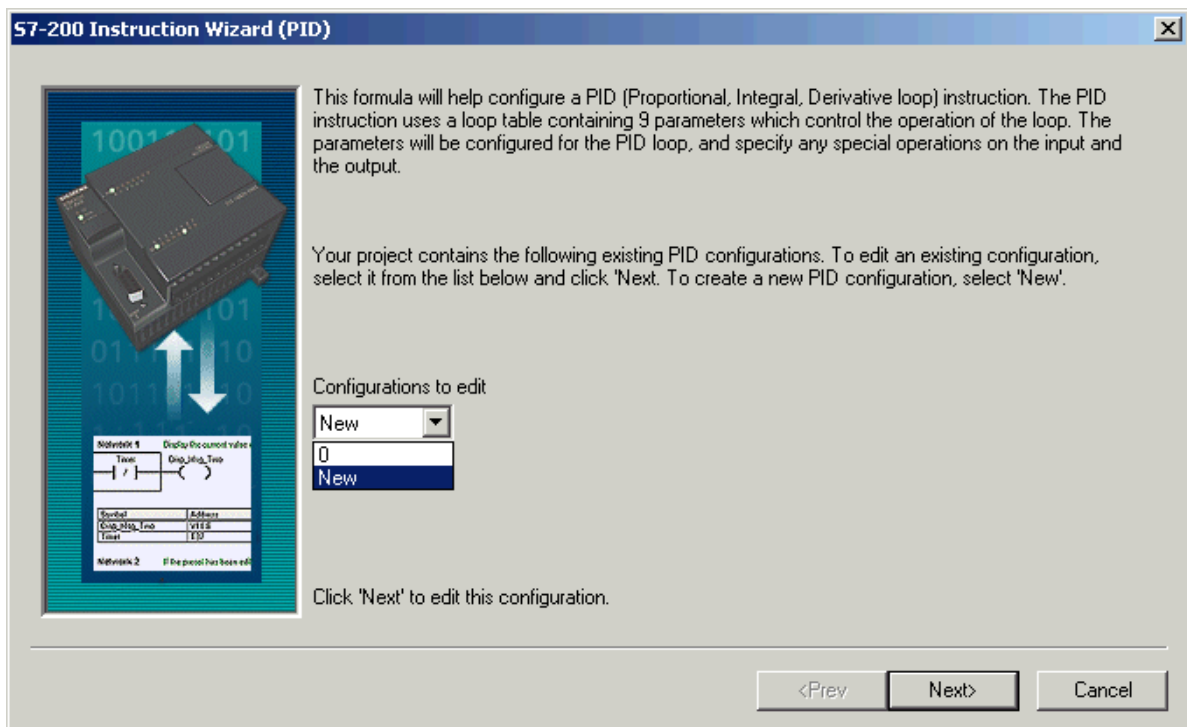


图 2. 选择需要配置的回路

第一步：定义需要配置的 PID 回路号

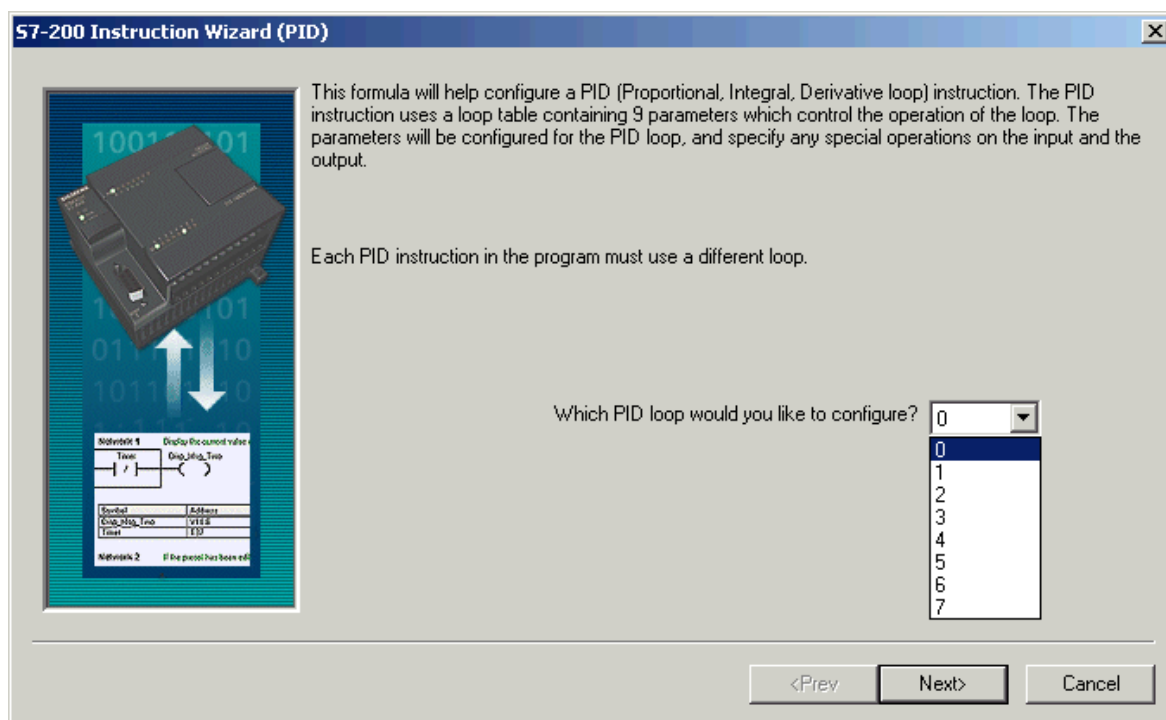


图 3. 选择 PID 回路号

第二步：设定 PID 回路参数

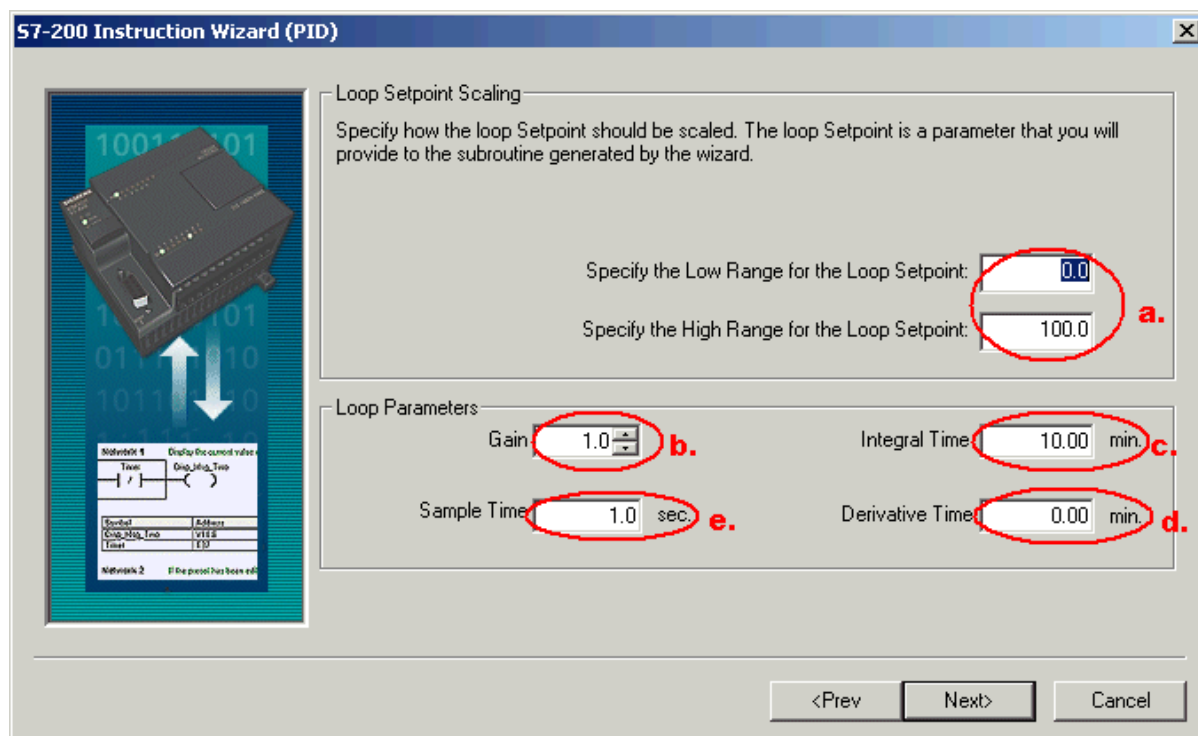


图 4. 设置 PID 参数

图 4 中：

a. 定义回路设定值（SP，即给定）的范围：

在低限（Low Range）和高限（High Range）输入域中输入实数，缺省值为 0.0 和 100.0，表示给定值的取值范围占过程反馈量程的百分比。

这个范围是给定值的取值范围。它也可以用实际的工程单位数值表示。参见：[设置给定—反馈的量程范围](#)。

以下定义 PID 回路参数，这些参数都应当是实数：

b. Gain（增益）：即比例常数。

c. Integral Time（积分时间）：如果不想要积分作用，可以把积分时间设为无穷大：9999.99

d. Derivative Time（微分时间）：如果不想要微分回路，可以把微分时间设为 0。

e. Sample Time（采样时间）：是 PID 控制回路对反馈采样和重新计算输出值的时间间隔。在向导完成后，若想要修改此数，则必须返回向导中修改，不可在程序中或状态表中修改。

注意：关于具体的 PID 参数值，每一个项目都不一样，需要现场调试来定，没有所谓经验参数。

第三步：设定回路输入输出值

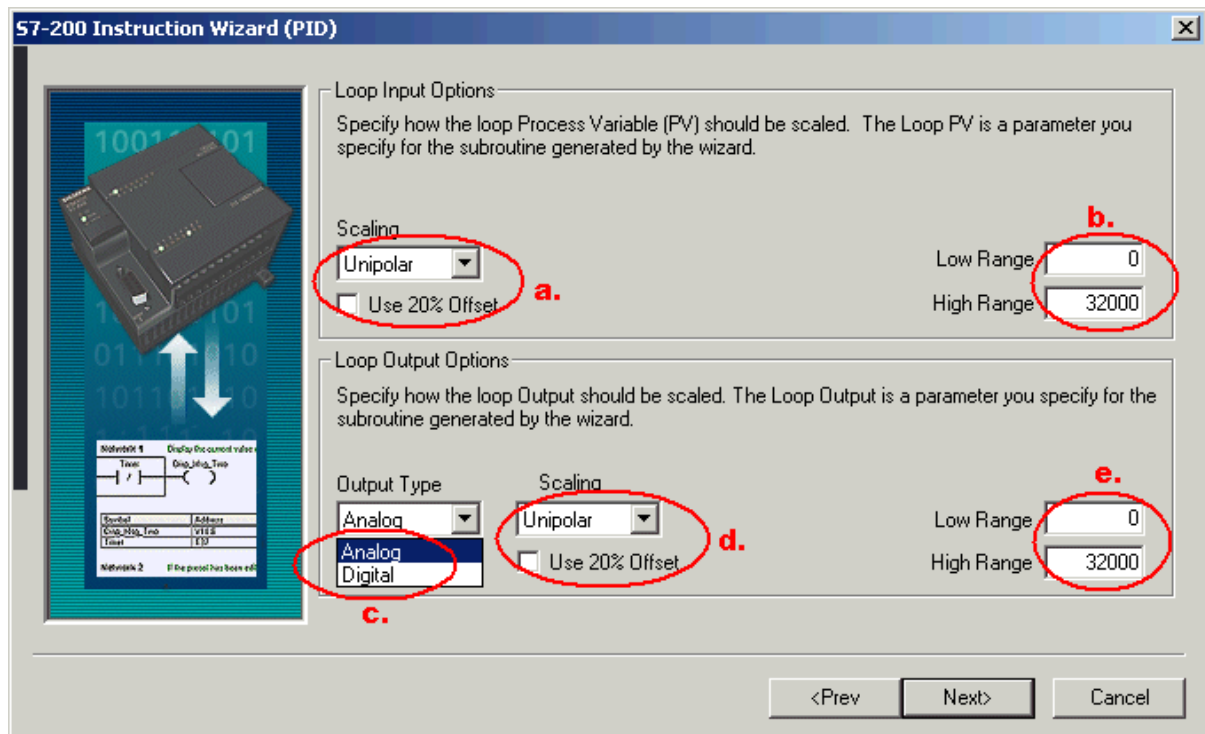


图 5. 设定 PID 输入输出参数

在图 5 中，首先 设定过程变量的范围：

a. 指定输入类型

- **Unipolar:** 单极性，即输入的信号为正，如 0—10V 或 0—20mA 等
- **Bipolar:** 双极性，输入信号在从负到正的范围内变化。如输入信号为 ±10V、±5V 等时选用
- **20% Offset:** 选用 20% 偏移。如果输入为 4—20mA 则选单极性及此项，4mA 是 0—20mA 信号的 20%，所以选 20% 偏移，即 4mA 对应 6400，20mA 对应 32000

b. 反馈输入取值范围

- 在 a. 设置为 Unipolar 时，缺省值为 0 - 32000，对应输入量程范围 0 - 10V 或 0 - 20mA 等，输入信号为正
- 在 a. 设置为 Bipolar 时，缺省的取值为 -32000 - +32000，对应的输入范围根据量程不同可以是 ±10V、±5V 等
- 在 a. 选中 20% Offset 时，取值范围为 6400 - 32000，不可改变

此反馈输入也可以是工程单位数值，参见：设置给定—反馈的量程范围。

然后定义输出类型

c. **Output Type**（输出类型）

可以选择模拟量输出或数字量输出。模拟量输出用来控制一些需要模拟量给定的设备，如比例阀、变频器等；数字量输出实际上是控制输出点的通、断状态按照一定的占空比变化，可以控制固态继电器（加热棒等）

d. 选择模拟量则需设定回路输出变量值的范围，可以选择：

- **Unipolar:** 单极性输出，可为 0—10V 或 0—20mA 等
- **Bipolar:** 双极性输出，可为正负 10V 或正负 5V 等
- **20% Offset:** 如果选中 20% 偏移，使输出为 4 - 20mA

e. 取值范围：

- d 为 Unipolar 时，缺省值为 0 到 32000
- d 为 Bipolar 时，取值 -32000 到 32000
- d 为 20% Offset 时，取值 6400 - 32000，不可改变

如果选择了开关量输出，需要设定此占空比的周期。

第四步：设定回路报警选项

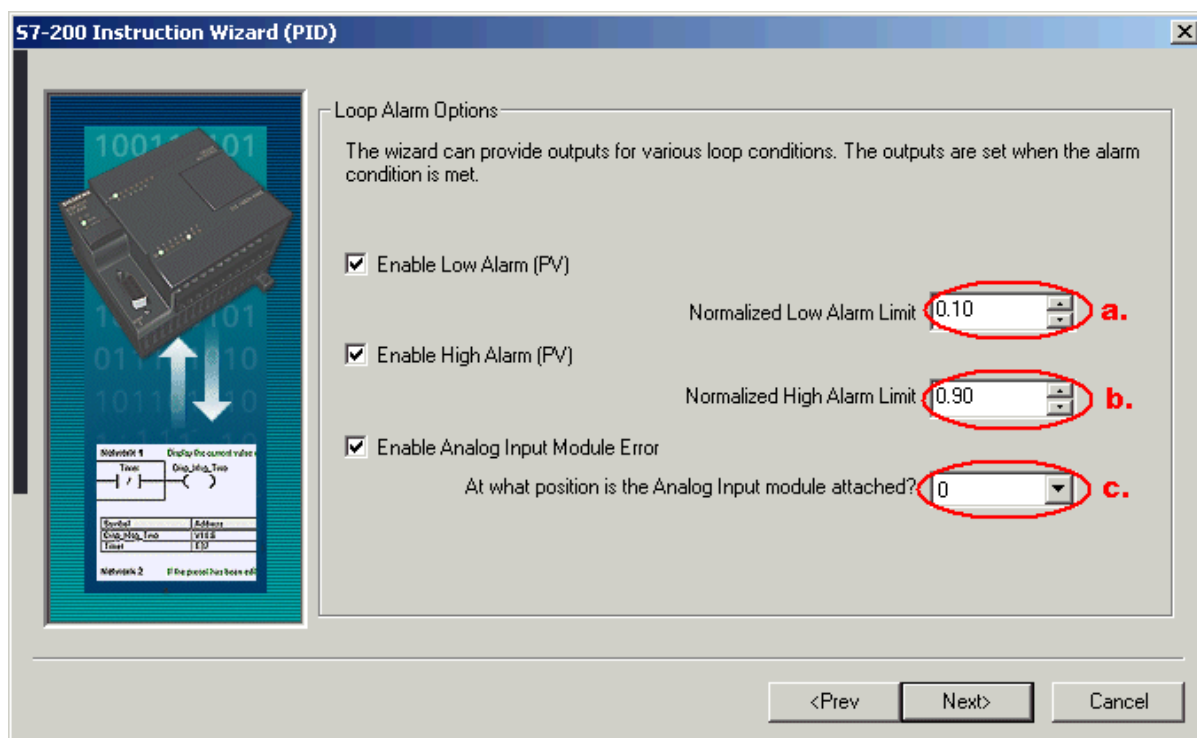


图 6. 设定回路报警限幅值

向导提供了三个输出来反映过程值 (PV) 的低值报警、高值报警及过程值模拟量模块错误状态。当报警条件满足时，输出置位为 1。这些功能在选中了相应的选择框之后起作用。

a. 使能低值报警并设定过程值 (PV) 报警的低值，此值为过程值的百分数，缺省值为 0.10，即报警的低值为过程值的 10%。此值最低可设为 0.01，即满量程的 1%

b. 使能高值报警并设定过程值 (PV) 报警的高值，此值为过程值的百分数，缺省值为 0.90，即报警的高值为过程值的 90%。此值最高可设为 1.00，即满量程的 100%

c. 使能过程值 (PV) 模拟量模块错误报警并设定模块于 CPU 连接时所处的模块位置。“0”就是第一个扩展模块的位置

第五步：指定 PID 运算数据存储区

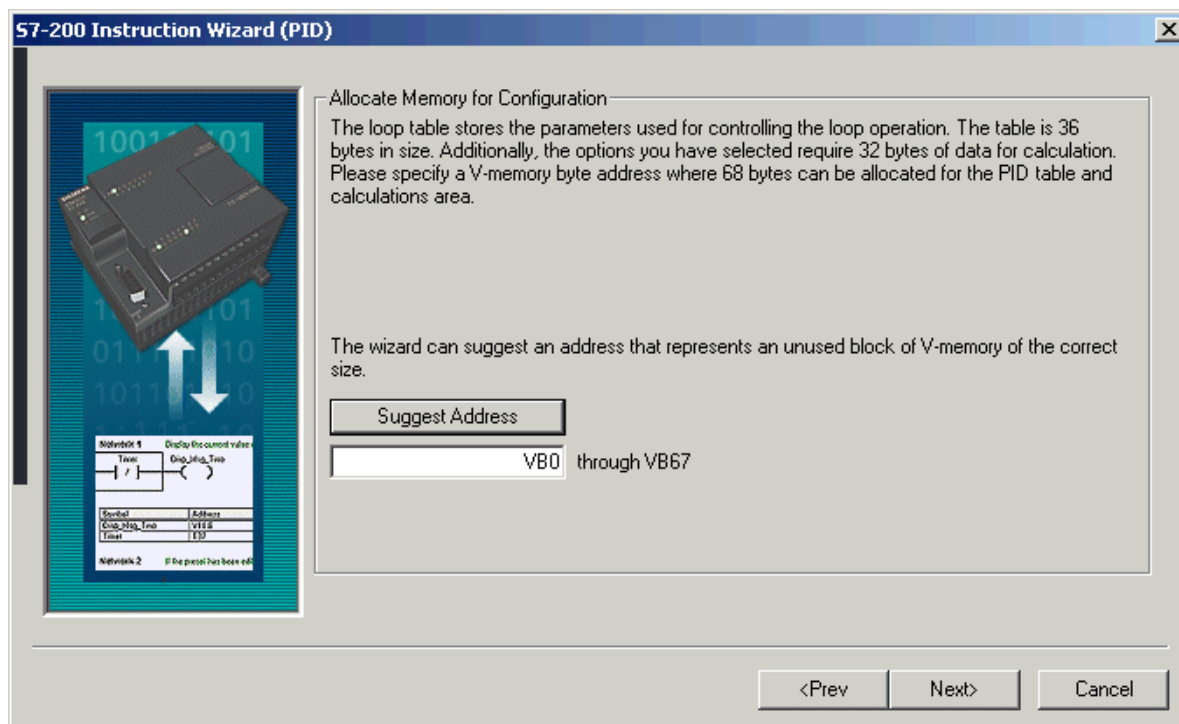


图 7. 分配运算数据存储区

PID 指令（功能块）使用了一个 120 个字节的 V 区参数表来进行控制回路的运算工作；除此之外，PID 向导生成的输入/输出量的标准化程序也需要运算数据存储区。需要为它们定义一个起始地址，要保证该地址起始的若干字节在程序的其它地方没有被重复使用。如果点击“Suggest Address”，则向导将自动为你设定当前程序中没有用过的 V 区地址。

自动分配的地址只是在执行 PID 向导时编译检测到空闲地址。向导将自动为该参数表分配符号名，用户不要再自己为这些参数分配符号名，否则将导致 PID 控制不执行。

第六步：定义向导所生成的 PID 初始化子程序和中断程序名及手/自动模式

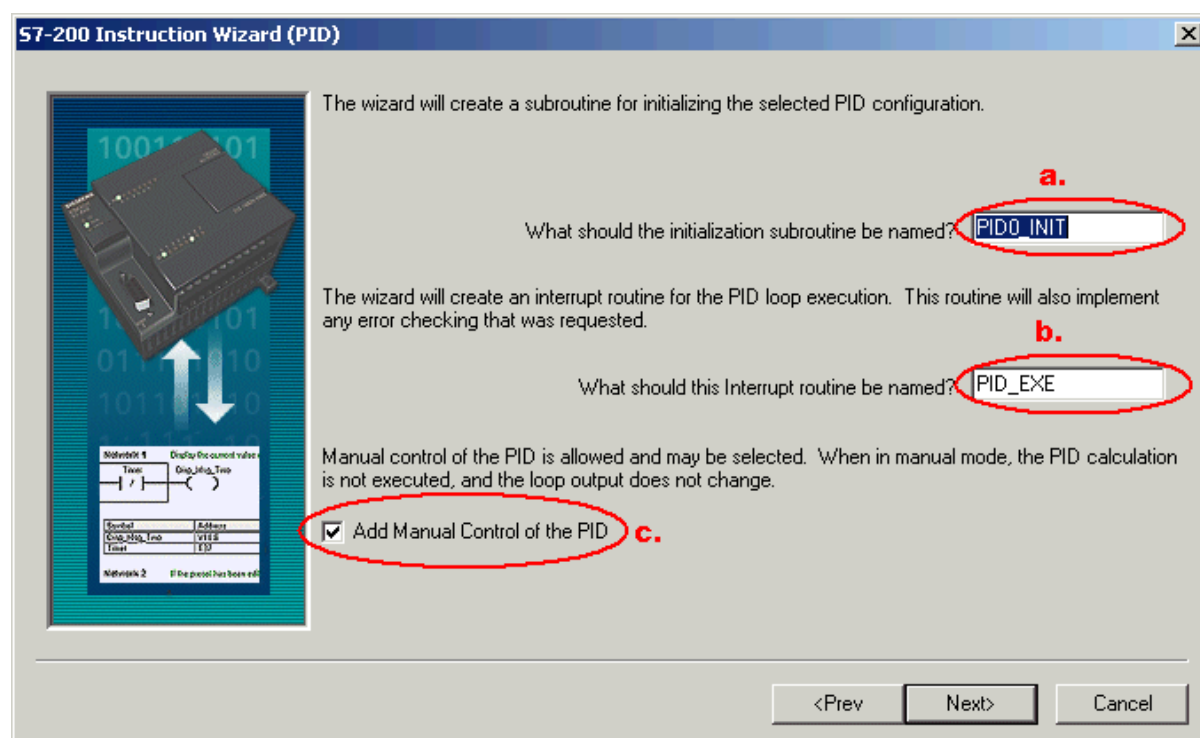


图 8. 指定子程序、中断服务程序名和选择手动控制

向导已经为初始化子程序和中断子程序定义了缺省名，你也可以修改成自己起的名字。

- a. 指定 PID 初始化子程序的名字。
- b. 指定 PID 中断子程序的名字

注意：

1. 如果你的项目中已经存在一个 PID 配置，则中断程序名为只读，不可更改。因为一个项目中所有 PID 共用一个中断程序，它的名字不会被任何新的 PID 所更改。
 2. PID 向导中断用的是 SMB34 定时中断，在用户使用了 PID 向导后，注意在其它编程时不要再用此中断，也不要向 SMB34 中写入新的数值，否则 PID 将停止工作。
- c. 此处可以选择添加 PID 手动控制模式。在 PID 手动控制模式下，回路输出由手动输出设定控制，此时需要写入手动控制输出参数一个 0.0—1.0 的实数，代表输出的 0%—100%而不是直接去改变输出值。

此功能提供了 PID 控制的手动和自动之间的无扰切换能力。

第七步：生成 PID 子程序、中断程序及符号表等

一旦点击完成按钮，将在你的项目中生成上述 PID 子程序、中断程序及符号表等。

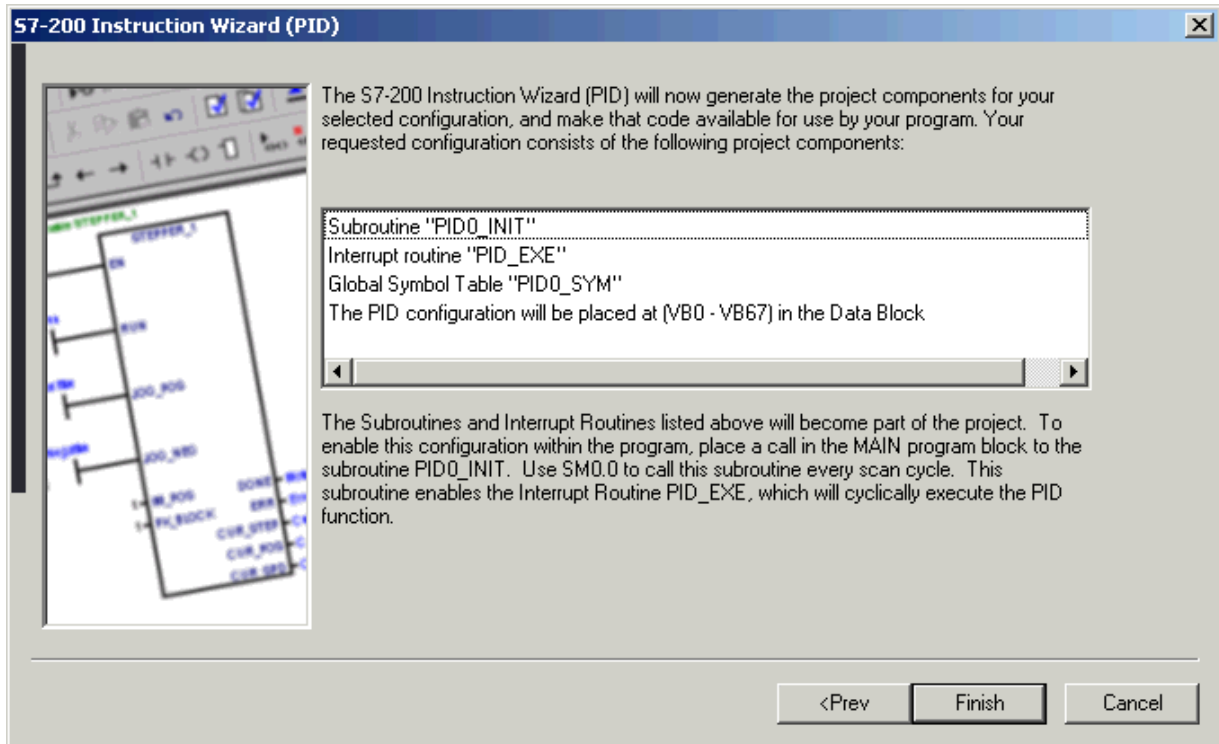


图 9. 生成 PID 子程序、中断程序和符号表等

第八步：配置完 PID 向导，需要在程序中调用向导生成的 PID 子程序（如下图）



图 10. PID 子程序

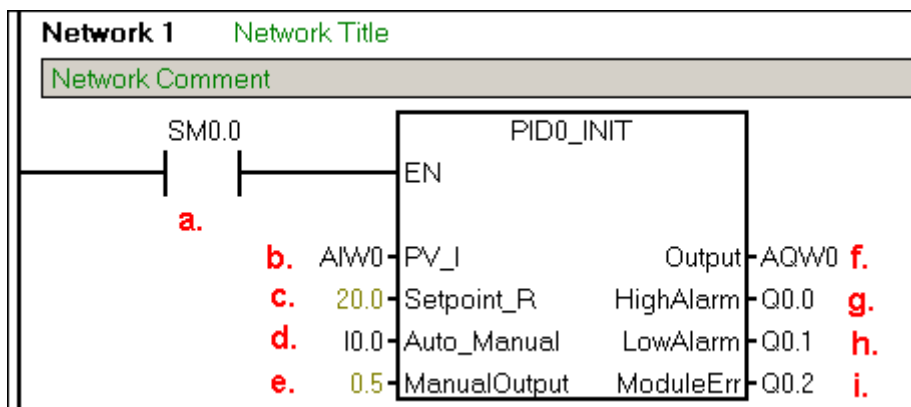


图 11. 调用 PID 子程序

在用户程序中调用 PID 子程序时，可在指令树的 Program Block（程序块）中用鼠标双击由向导生成的 PID 子程序，在局部变量表中，可以看到有关形式参数的解释和取值范围。

- a. 必须用 SM0.0 来使能 PID，以保证它的正常运行
- b. 此处输入过程值（反馈）的模拟量输入地址
- c. 此处输入设定值变量地址（VDxx），或者直接输入设定值常数，根据向导中的设定 0.0—100.0，此处应输入一个 0.0—100.0 的实数，例：若输入 20，即为过程值的 20%，假设过程值 AIW0 是量程为 0—200 度的温度值，则此处的设定值 20 代表 40 度（即 200 度的 20%）；如果在向导中设定给定范围为 0.0 - 200.0，则此处的 20 相当于 20 度
- d. 此处用 IO.0 控制 PID 的手/自动方式，当 IO.0 为 1 时，为自动，经过 PID 运算从 AQW0 输出；当 IO.0 为 0 时，PID 将停止计算，AQW0 输出为 ManualOutput（VD4）中的设定值，此时不要另外编程或直接给 AQW0 赋值。若在向导中没有选择 PID 手动功能，则此项不会出现
- e. 定义 PID 手动状态下的输出，从 AQW0 输出一个满值范围内对应此值的输出量。此处可输入手动设定值的变量地址（VDxx），或直接输入数。数值范围为 0.0—1.0 之间的一个实数，代表输出范围的百分比。例：如输入 0.5，则设定为输出的 50%。若在向导中没有选择 PID 手动功能，则此项不会出现
- f. 此处键入控制量的输出地址
- g. 当高报警条件满足时，相应的输出置位为 1，若在向导中没有使能高报警功能，则此项将不会出现
- h. 当低报警条件满足时，相应的输出置位为 1，若在向导中没有使能低报警功能，则此项将不会出现
- i. 当模块出错时，相应的输出置位为 1，若在向导中没有使能模块错误报警功能，则此项将不会出现

调用 PID 子程序时，不用考虑中断程序。子程序会自动初始化相关的定时中断处理事项，然后中断程序会自动执行。

第九步：实际运行并调试 PID 参数



没有一个 PID 项目的参数不需要修改而能直接运行，因此需要在实际运行时调试 PID 参数。

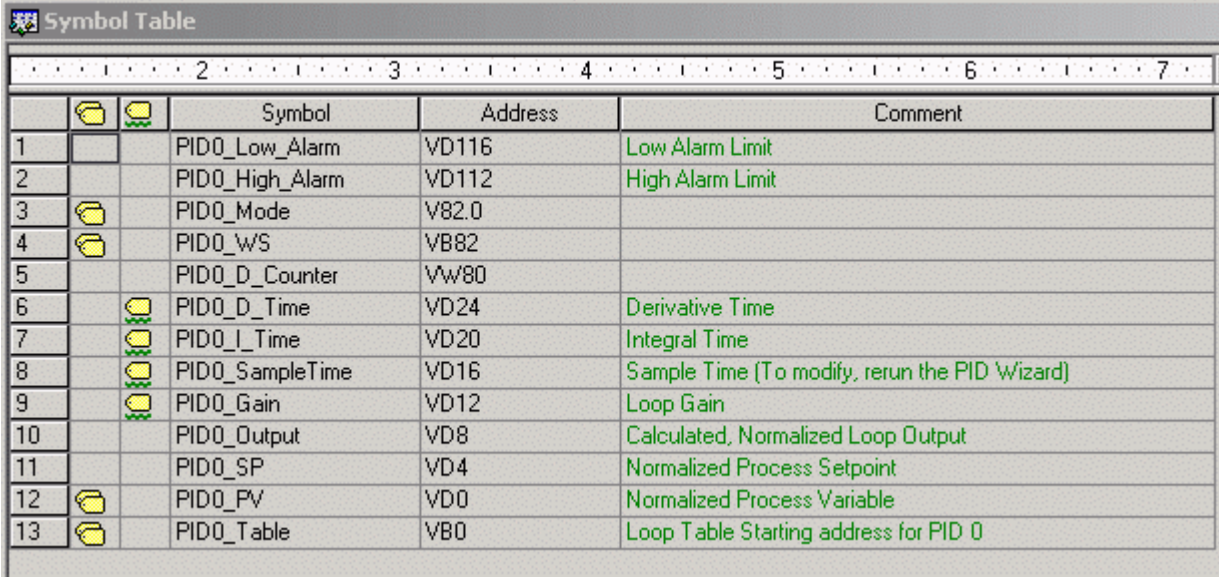
查看 Data Block（数据块），以及 Symbol Table（符号表）相应的 PID 符号标签的内容，可以找到包括 PID 核心指令所用的控制回路表，包括比例系数、积分时间等等。将此表的地址复制到 Status Chart（状态表）中，可以在监控模式下在线修改 PID 参数，而不必停机再次做组态。

参数调试合适后，用户可以在数据块中写入，也可以再做一次向导，或者编程向相应的数据区传送参数。

常问问题

1、做完 PID 向导后，如何知道向导中设定值，过程值及 PID 等参数所用的地址？


- 做完 PID 向导后可在 Symbol Table（符号表）中，查看 PID 向导所生成的符号表(上例中为 PID0_SYM)，可看到各参数所用的详细地址，及数值范围。
- 在 Data Block（数据块）中，查看 PID 指令回路表的相关参数。如图所示：



		Symbol	Address	Comment
1		PID0_Low_Alarm	VD116	Low Alarm Limit
2		PID0_High_Alarm	VD112	High Alarm Limit
3		PID0_Mode	V82.0	
4		PID0_WS	V882	
5		PID0_D_Counter	VW80	
6		PID0_D_Time	VD24	Derivative Time
7		PID0_I_Time	VD20	Integral Time
8		PID0_SampleTime	VD16	Sample Time (To modify, rerun the PID Wizard)
9		PID0_Gain	VD12	Loop Gain
10		PID0_Output	VD8	Calculated, Normalized Loop Output
11		PID0_SP	VD4	Normalized Process Setpoint
12		PID0_PV	VD0	Normalized Process Variable
13		PID0_Table	V80	Loop Table Starting address for PID 0

图 12. PID 数据块

2、做完 PID 向导后，如何在调试中修改 PID 参数？

可以在 Status Chart（状态表）中，输入相应的参数地址，然后在线写入用户需要的 PID 参数数值，这样用户就可根据工艺需要随时对 PID 参数、设定值等进行调整。

3、PID 已经调整合适，如何正式确定参数？

可以在 Data Block（数据块）中直接写入参数。

4、做完 PID 向导后，能否查看 PID 生成的子程序，中断程序？

PID 向导生成的子程序，中断程序用户是无法看到的，也不能对其进行修改。没有密码能够打开这些子程序，一般的应用也没有必要打开查看。

5、PID 参数有经验值吗？

每一个项目的 PID 参数都不一样，没有经验参数，只能现场调试获得。

6、我的 PID 向导生成的程序为何不执行？

- 必须保证用 SM0.0 无条件调用 PID0_INIT 程序
- 在程序的其它部分不要再使用 SMB34 定时中断，也不要对 SMB34 赋值

7、如何实现 PID 反作用调节？

在有些控制中需要 PID 反作用调节。例如：在夏天控制空调制冷时，若反馈温度（过程值）低于设定温度，需要关阀，减小输出控制（减少冷水流量等），这就是 PID 反作用调节（在 PID 正作用中若过程值小于设定值，则需要增大输出控制）。

若想实现 PID 反作用调节，需要把 PID 回路的增益设为负数。对于增益为 0 的积分或微分控制来说，如果指定积分时间、微分时间为负值，则是反作用回路。

8、如何根据工艺要求有选择地投入 PID 功能？

可使用“手动/自动”切换的功能。PID 向导生成的 PID 功能块只能使用 SM0.0 的条件调用。

PID Wizard 中的给定—反馈设置

完成 PID Wizard 组态后，会为每个 PID 回路生成一个子程序 PIDx_INIT

(x = 0 - 7)。在用户程序中，必须使用 SM0.0 始终调用这个子程序才能实现 PID 功能。

下图是一个最简单的 PID 子程序调用程序段：

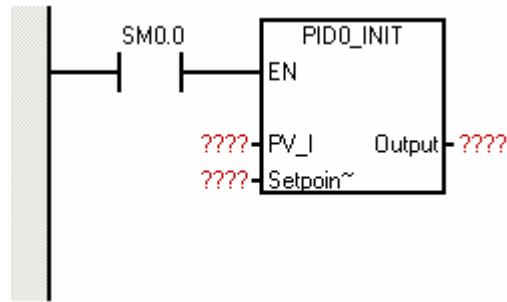


图 1. 调用 PID 子程序

其中：

- PV_I： 过程反馈参数值的入口
- Setpoint： 给定参数值的入口
- Output： PID 调节器的输出值

在这里，给定、反馈的入口参数不是 PID 指令功能块所需要的 0.0 - 1.0 之间的实数，而可以是实际的反馈地址，或是其他变量。例如，PV_I 可以是模拟量输入地址 AIW0，也可以是存储器地址 VW100 等；Setpoint 则往往来自 V 变量存储区，这样就可以从人机操作界面（HMI）设备输入给定值。

注意：

对于 PID 控制系统来说，必须保证给定与过程反馈的一致性：

- 给定与反馈的物理意义一致

这取决于被控制的对象，如果是压力，则给定也必须对应于压力值；如果是温度，则给定也必须对应于温度。

- 给定与反馈的数值范围对应

如果给定直接是摄氏温度值，则反馈必须是对应的摄氏温度值；如果反馈直接使用模拟量输入的对应数值，则给定也必须向反馈的数值范围换算。

如果给定与反馈的换算有特定的比例关系也可以。如给定也可以表示为以反馈的数值范围的百分比数值。

给定与反馈的数值具体是什么数值，其取值范围究竟如何，完全取决于我们在使用“PID 向导”编程时指定的给定与反馈的数值范围。其中，反馈量的数值范围不能随便自己定义，而要取决于具体应用的模拟量输入模块。

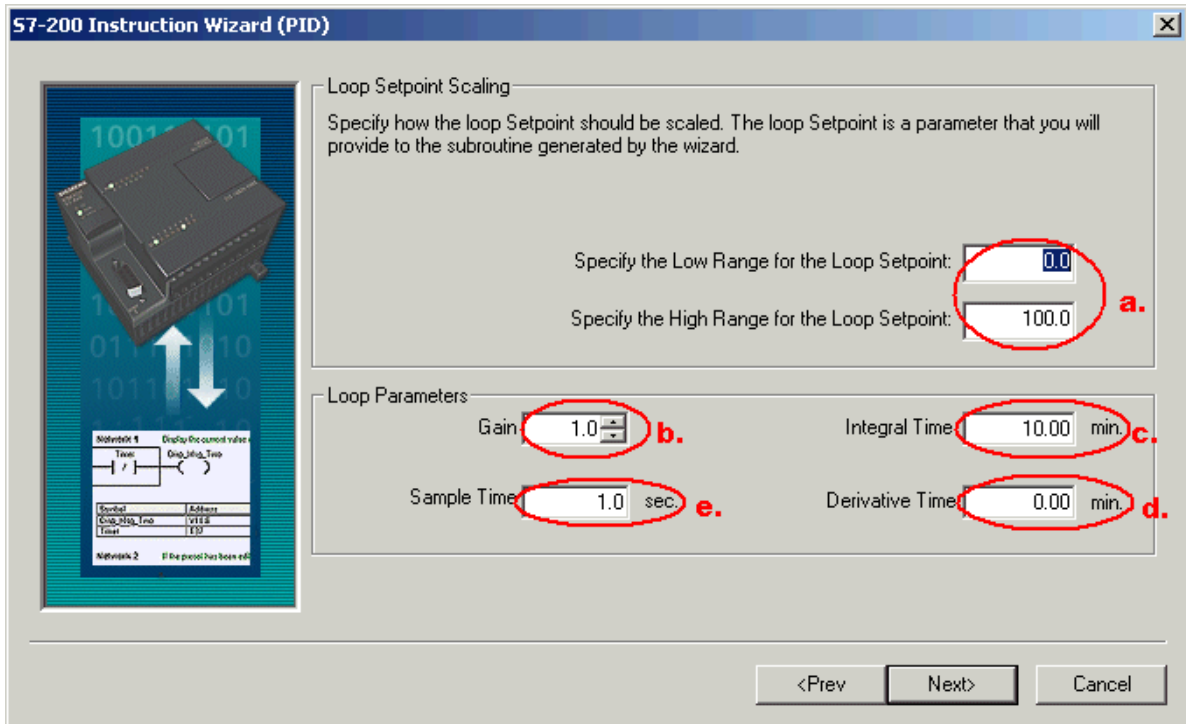


图 2. 在图中 a. 处设置给定范围

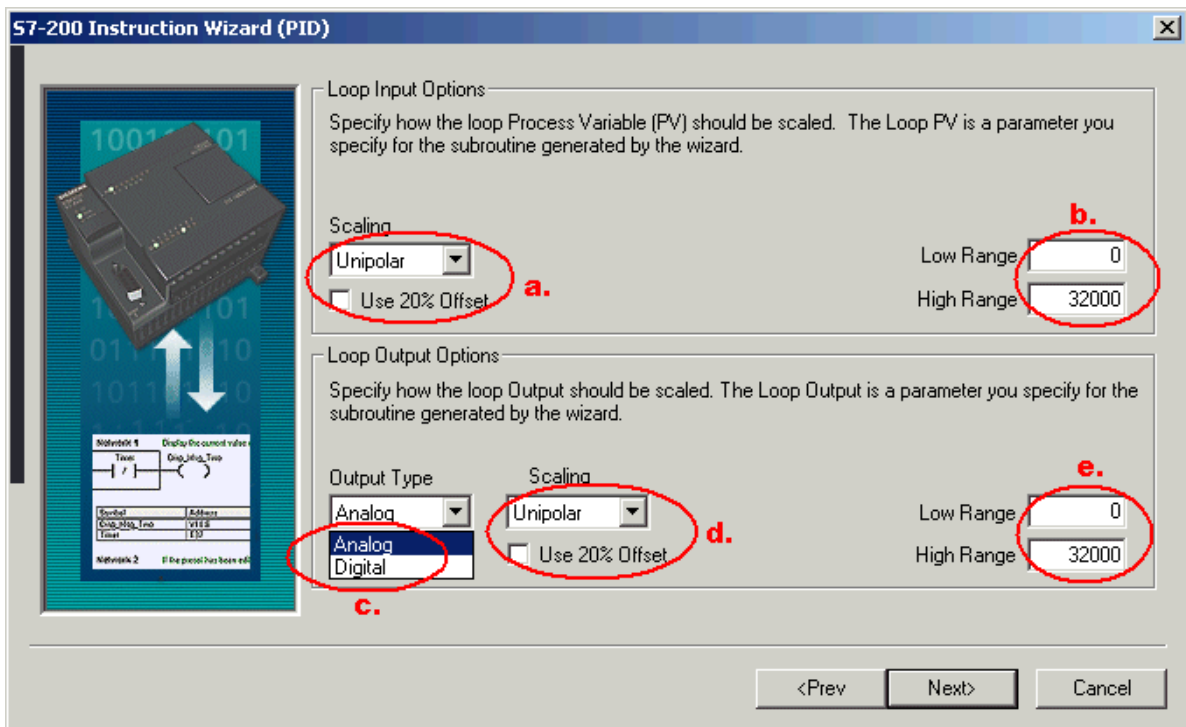


图 3. 在图中 b. 处设置反馈范围

实例

假定一个 PID 控制系统的控制对象是压力，反馈元件的测量范围为 0 - 16MPa。反馈器件的信号经过变换，以 0 - 20mA（或 4 - 20mA）电流信号的形式输入到 EM231 模拟量输入模块中。据此，我们可以按下表设置给定、反馈的范围。

表 1.

	反馈（单极性）		给定	
	实际物理量	模拟量输入数值	百分比形式（占 0 - 16MPa 的百分比）	物理工程单位形式
高限	16 MPa	32000	100.0	$n \times 16.0$
低限	0 MPa	0 (0 - 20mA)	0.0	0.0
		6400 (4 - 20mA)		

n 为比例系数，为了精度高些可以设置 $n=10$ 等等

又如一个温度控制的 PID 系统，温度值直接由热电偶测量，输入到 EM231 TC（热电偶）模块转换为温度值。热电偶为 J 型，其测量范围为 $-150.0^{\circ}\text{C} - 1200.0^{\circ}\text{C}$ 。则可按如下设置给定的范围。

表 2.

	反馈（双极性）		给定	
	实际物理量	模拟量输入数值	百分比形式（占 $-150^{\circ}\text{C} - 1200.0^{\circ}\text{C}$ 的百分比）	物理工程单位形式
高限	1200.0°C	12000	100.0	1200
低限	-150.0°C	-1500	0.0	-150

在上面的例子中，反馈和给定可以按照如下方法设置

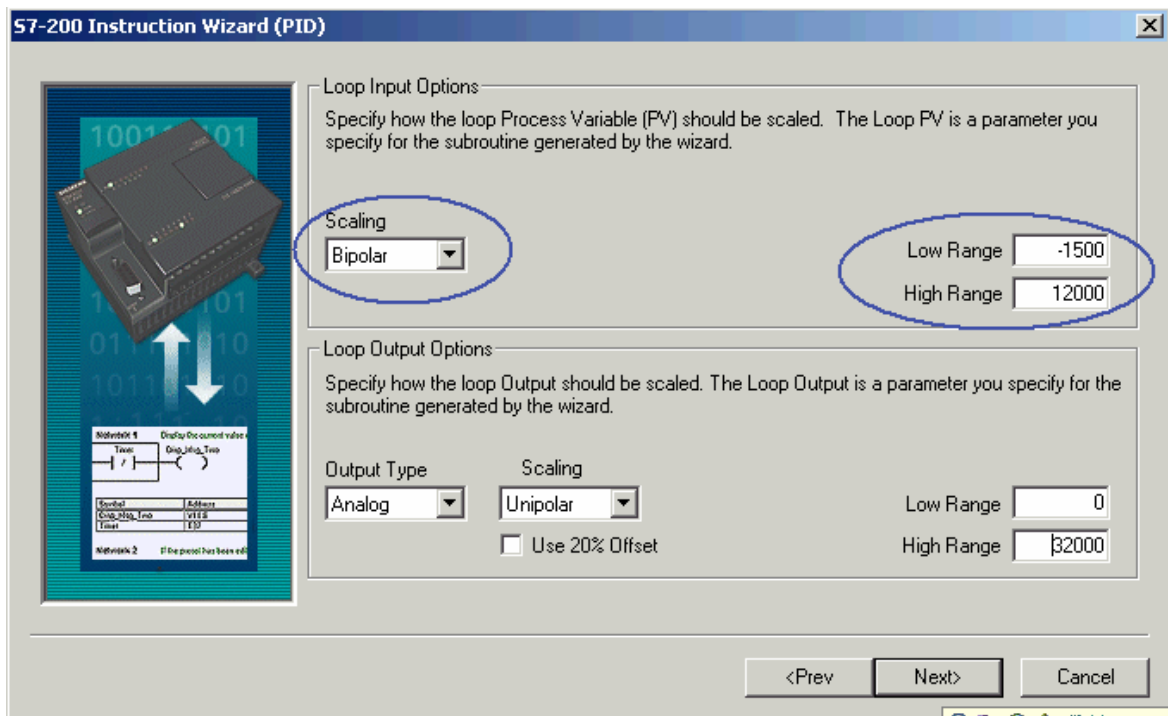


图 4. 反馈范围设置

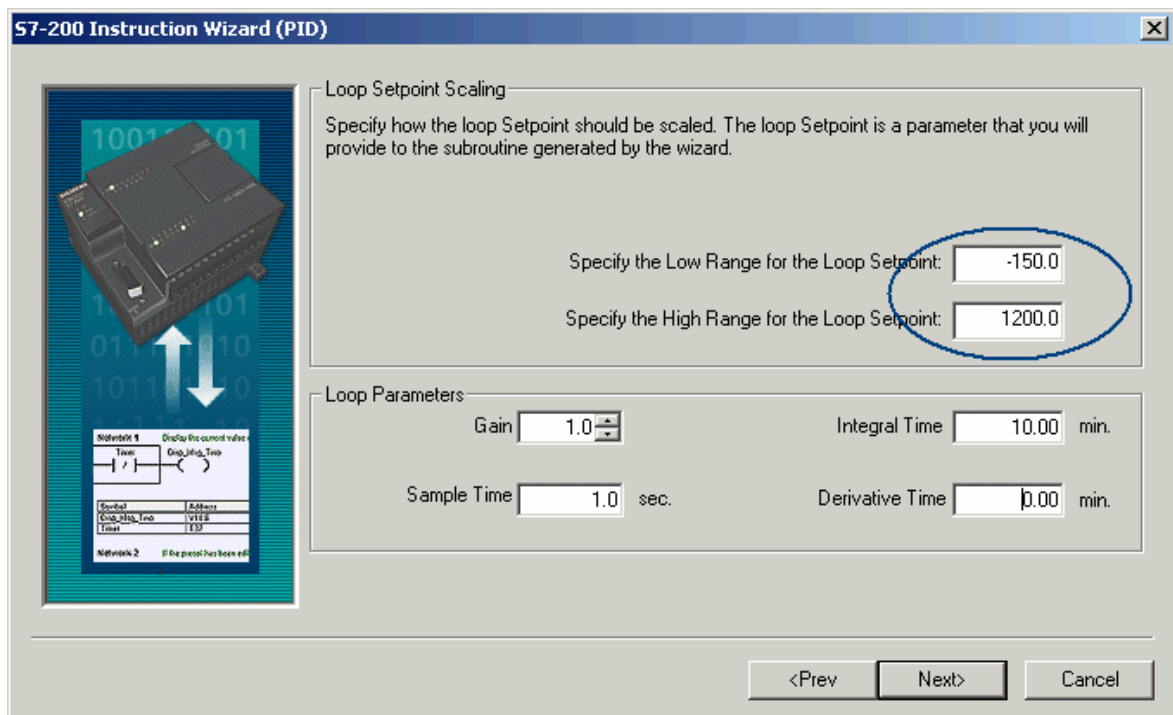


图 5. 给定范围设置

第二部分、PID 自整定

PID 自整定

新的 S7-200 CPU 支持 PID 自整定功能，在 STEP 7-Micro/WIN V4.0 中也添加了 PID 调节控制面板。

用户可以使用用户程序或 PID 调节控制面板来启动自整定功能。在同一时间最多可以有 8 个 PID 回路同时进行自整定。PID 调节控制面板也可以用来手动调试老版本的（不支持 PID 自整定）CPU 的 PID 控制回路。

用户可以根据工艺要求为调节回路选择快速响应、中速响应、慢速响应或极慢速响应。PID 自整定会根据响应类型而计算出最优化的比例、积分、微分值，并可应用到控制中。

PID 调节控制面板

STEP 7-Micro/WIN V4.0 中提供了一个 PID 调节控制面板，可以用图形方式监视 PID 回路的运行，另外从面板中还可以启动、停止自整定功能。

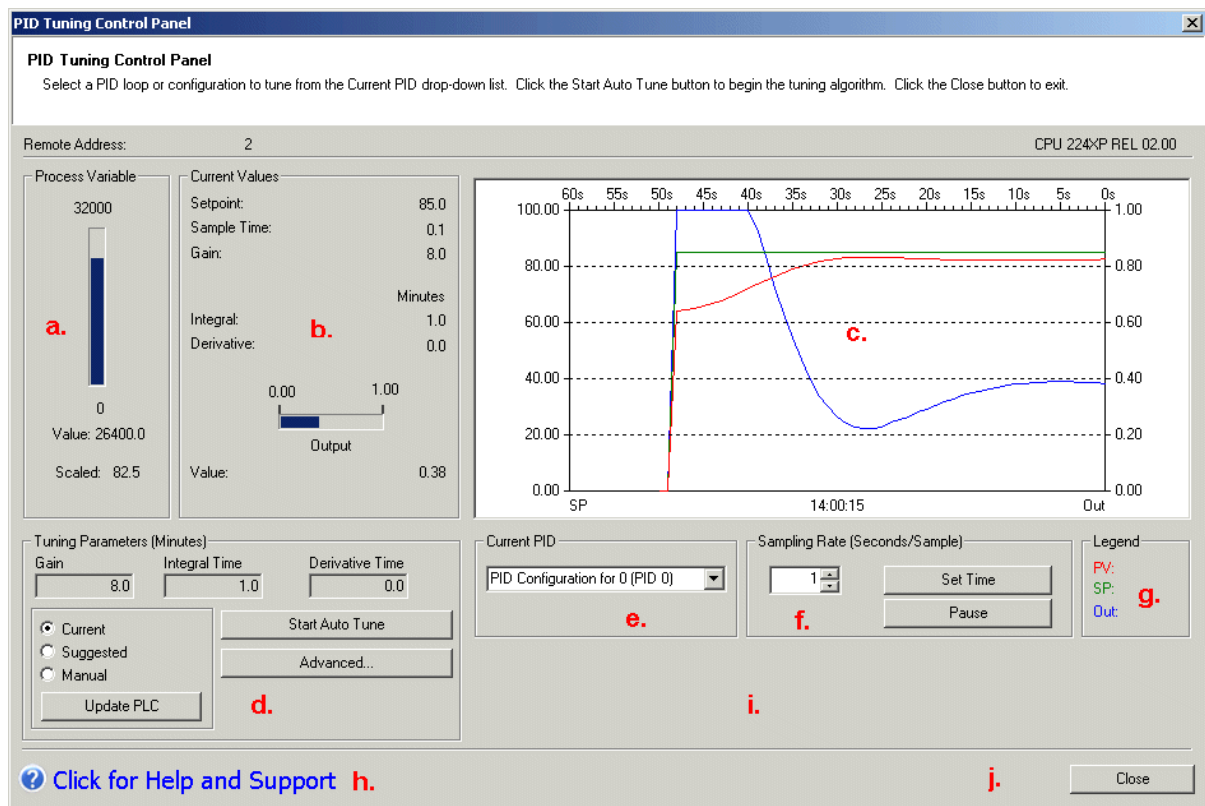


图 1. PID 调节控制面板

在图 1 中：

a. 过程值指示

显示过程变量的值及其棒图

b. 当前的输出值指示

显示当前使用的设定值、采样时间、PID 参数值及显示当前的输出值和棒图

c. 可显示过程值、设定值及输出值的 PID 趋势图



图 2. 图形显示区

图中：

A. 过程变量和设定值的取值范围及刻度

B. PID 输出的取值范围及刻度

C. 实际 PC 时间

D. 以不同颜色表示的设定值、过程变量及输出的趋势图

d. 调节参数

这里你可以：

- 选择 PID 参数的显示：当前参数 (Current)、推荐参数 (Suggested)、手动输入 (Manual)
- 在 Manual 模式下，可改变 PID 参数，并按 Update PLC 按钮来更新 PLC 中的参数
- 启动 PID 自整定功能
- 选择 Advanced (高级) 按钮进入高级参数设定

e. 当前的 PID 回路号

这里你可以选择需要监视或自整定的 PID 回路

f. 时间选项设定

这里你可以设定趋势图的时基，时基以分为单位

g. 图例颜色

这里你可以看到趋势图中不同的颜色代表不同的值的趋势

- h. 帮助按钮
- i. PID 信息显示窗口
- j. 关闭 PID 调节面板

要使用 PID 调节控制面板，PID 编程必须使用 PID 向导完成。



PID 自整定步骤

第一步：在 PID Wizard（向导）中完成 PID 功能组态

要想使用 PID 自整定功能，PID 编程必须用 PID 向导来完成

第二步：打开 PID 调节控制面板，设置 PID 回路调节参数

在 Micro/WIN V4.0 在线的情况下，从主菜单 Tools > PID Tune Control Panel 或点

 击  进入 PID 调节控制面板中，如果面板没有被激活（所有地方都是灰色），可点击 Configure（配置）按钮运行 CPU。

在 PID 调节面板的 e. 区选择要调节的 PID 回路号，在 d. 区选择 Manual（手动），调节 PID 参数并点击 Update（更新），使新参数值起作用，监视其趋势图，根据调节状况改变 PID 参数直至调节稳定。

为了使 PID 自整定顺利进行，应当做到：

- 使 PID 调节器基本稳定，输出、反馈变化平缓，并且使反馈比较接近给定
- 设置合适的给定值，使 PID 调节器的输出远离趋势图的上、下坐标轴，以免 PID 自整定开始后输出值的变化范围受限制

参见：手动调整 PID 回路参数

第三步：在 d. 区点击 Advanced（高级）按钮，设定 PID 自整定选项。如果不是很特殊的系统，也可以不加理会。

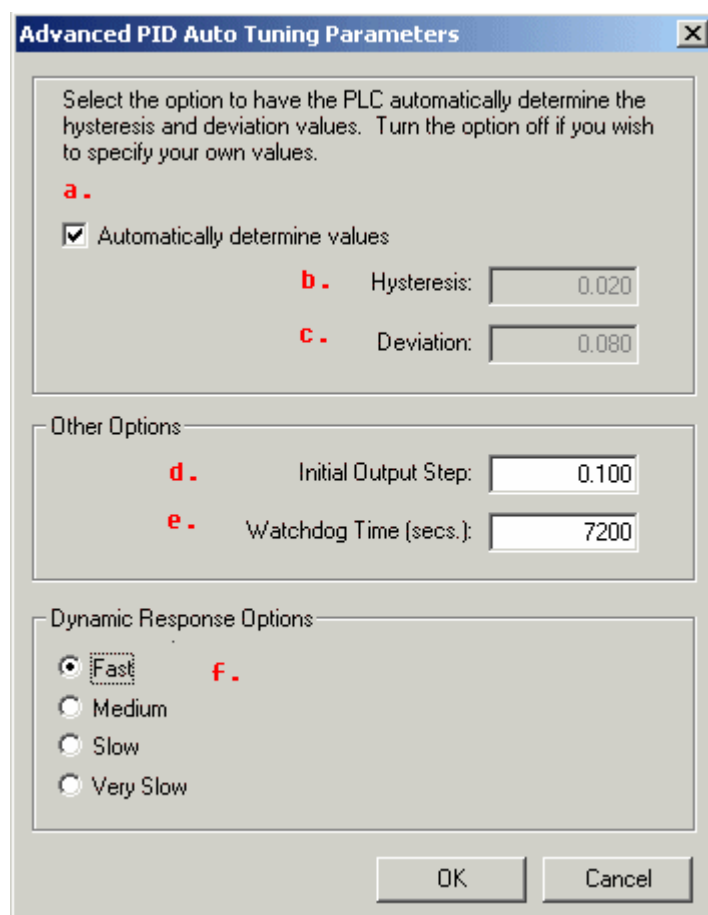


图 3. 设置 PID 自整定高级选项

在此允许你设定下列参数：

a. 你可以选中复选框，让自整定来自动计算死区值和偏移值

对于一般的 PID 系统，建议使用自动选择。

b. Hysteresis（滞回死区）：

死区值规定了允许过程值偏离设定值的最大（正负）范围，过程反馈在这个范围内的变化不会引起 PID 自整定调节器改变输出，或者使 PID 自整定调节器“认为”这个范围内的变化是由于自己改变输出进行自整定调节而引起的。PID 自整定开始后，只有过程反馈值超出了该区域，PID 自整定调节器才会认为它对输出的改变发生了效果。这个值用来减少过程变量的噪声对自整定的干扰，从而更精确地计算出过程系统的自然振动频率。如果选用自动计算，则缺省值为 2%。

如果过程变量反馈干扰信号较强（噪声大）自然变化范围就大，可能需要人为设置一个较大的值。但这个值的改变要与下面的偏差值保持 1:4 的关系。

c. Deviation（偏差）：

偏差值决定了允许过程变量偏离设定值的峰峰值。如果选择自动计算该值，它将是死区的 4 倍，即 8%。

有些非常敏感的系统不允许过程量偏离给定值很多，也可以人工设置为比较小的值，但是要和上述“死区”设置保持比例关系。这就是说，一个精度要求高的系统，其反馈信号必须足够稳定。

d. Initial Output Step（初始步长值）：PID 调节的初始输出值

PID 自整定开始后，PID 自整定调节器将主动改变 PID 的输出值，以观察整个系统的反应。初始步长值就是输出的变动第一步变化值，以占实际输出量程的百分比表示。

e. Watchdog Time（看门狗时间）：过程变量必须在此时间（时基为秒）内达到或穿越给定值，否则会产生看门狗超时错误。

PID 自整定调节器在改变输出后，如果超过此时间还未观察到过程反馈（从下至上或从上至下）穿越给定曲线，则超时。如果能够事先确定实际系统响应非常慢，可以加长这个时间。

f. 动态响应选项：根据回路过程（工艺）的要求可选择不同的响应类型：快速、中速、慢速、极慢速

- 快速：可能产生超调，属于欠阻尼响应
- 中速：在产生超调的边缘，属于临界阻尼响应
- 慢速：不会产生任何超调，属于过阻尼响应
- 极慢速：不会产生任何超调，属于严重过阻尼响应

用户在这里指定需要达到的系统控制效果，而不是对系统本身响应快慢的判断。

g. 设定完参数点击 OK 键回到 PID 调节控制面板的主画面

第四步：在手动将 PID 调节到稳定状态后，即过程值与设定值接近，且输出没有不规则的变化，并最好处于控制范围中心附近。此时可点击 d. 区内的 Start Auto Tune 按钮启动 PID 自整定功能，这时按钮变为 Stop Auto Tune。这时只需耐心等待，系统完成自整定后会自动将计算出的 PID 参数显示在 d. 区。当按钮再次变为 Start Auto Tune 时，表示系统已经完成了 PID 自整定。

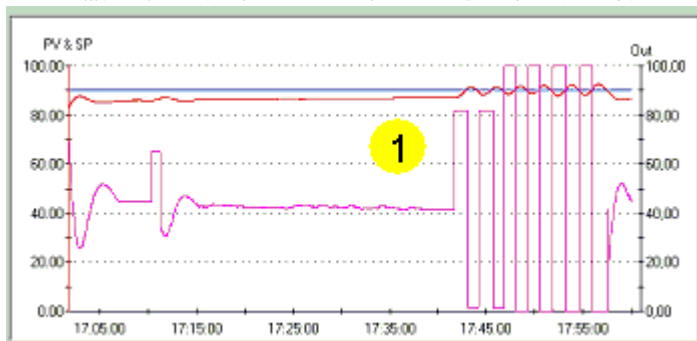
要使用自整定功能，必须保证 PID 回路处于自动模式。开始自整定后，给定值不能再改变。

第五步：如果用户想将 PID 自整定的参数应用到当前 PLC 中，则只需点击 Update PLC。

完成 PID 调整后，最好下载一次整个项目（包括数据块），使新参数保存到 CPU 的 EEPROM 中。

PID 自整定失败的原因

1. PID 输出在最大值与最小值之间振荡（曲线接触到坐标轴）



解决方法：降低 PID 初始输出步长值（initial output step）

2. 经过一段时间后，PID 自整定面板显示如下信息：“The Auto Tune algorithm was aborted due to a zero-crossing watchdog timeout.” 即自整定计算因为等待反馈穿越给定值的看门狗超时而失败。

解决方法：确定在启动 PID 自整定前，过程变量和输出值已经稳定。并检查 Watchdog Time 的值，将其适当增大。

对于其它错误，可参考手册中表 15-3 中的错误代码的描述。

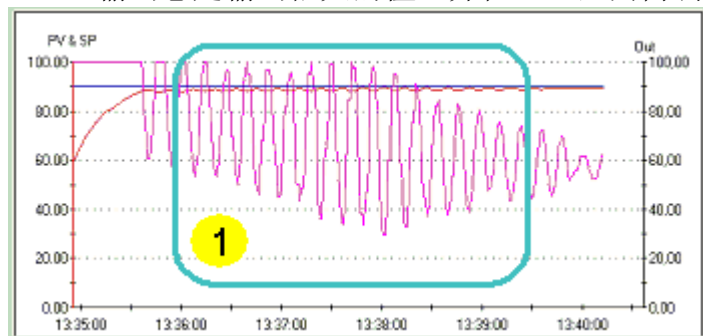
如何获得一个稳定的 PID 回路

在开始 PID 自整定调整前，整个 PID 控制回路必须工作在相对稳定的状态。

稳定的 PID 是指过程变量接近设定值，输出不会不规则的变化，且回路的输出值在控制范围中心附近变化。

问题与解决方法：

1. PID 输出总是输出很大的值，并在这一区间内调节变化

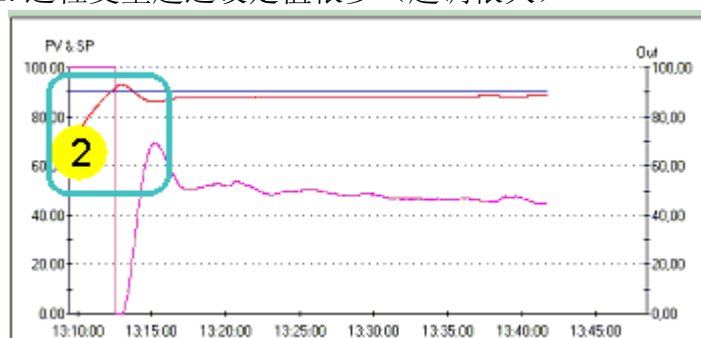


产生原因：

- 增益 (Gain) 值太高
- PID 扫描时间 (sample time) 太长 (对于快速响应 PID 的回路)

解决方法：降低增益 (Gain) 值并且/或选择短一些的扫描时间

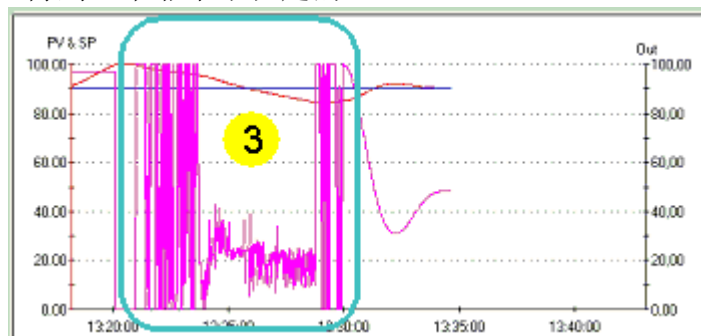
2. 过程变量超过设定值很多 (超调很大)



产生原因：积分时间 (Integral time) 可能太高

解决方法：降低积分时间

3. 得到一个非常不稳定的 PID



产生原因:

- 如果用了微分, 可能是微分参数有问题
- 没有微分, 可能是增益 (Gain) 值太高

解决方法:

- 调整微分参数到 0—1 的范围内
- 根据回路调节特性将增益值降低, 最低可从 0. x 开始逐渐增大往上调, 直到获得稳定的 PID。