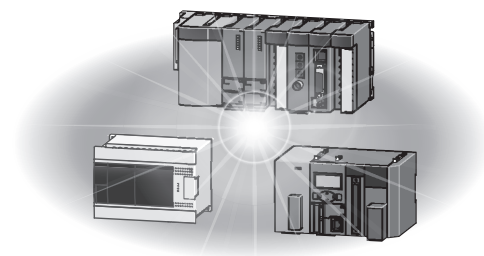


三菱电机 **通用** 可编程控制器

MELSEC **Q** series MELSEC *L* series

MELSEC-Q/L/QnA编程手册  
(PID控制指令篇)

---





## ● 安全注意事项 ●

(使用之前请务必阅读)

在使用本产品之前，应仔细阅读本手册及本手册中所介绍的关联手册，同时在充分注意安全的前提下正确地进行操作。

本手册中的注意事项仅记载了与本产品有关的内容，有关可编程控制器系统的安全守则，请阅读 CPU 模块用户手册。

在“安全注意事项”中，安全注意事项被分为“警告”和“注意”这二个等级。



警告

表示错误操作可能造成灾难性后果，引起死亡或重伤事故。



注意

表示错误操作可能造成危险的后果，引起人员中等伤害或轻伤还可能使设备损坏。

注意根据情况不同，即使△注意这一级别的事项也有可能引发严重后果。

对两级注意事项都须遵照执行，因为它们对于操作人员安全是至关重要的。

请妥善保管本手册，放置于操作人员易于取阅的地方，并应将本手册交给最终用户。

### [设计注意事项]

#### △ 警告

- 应在可编程控制器的外部配置安全电路，以保证在外部电源异常或可编程控制器本身故障时整个系统能向安全的方向动作。否则有可能导致误输出、误动作而发生事故。

(1) 应在可编程控制器的外部配置异常停止电路、保护电路、正转/反转等相反动作的互锁电路、定位的上限/下限等防止机械损坏的互锁电路等。

(2) 当可编程控制器检测到下列故障时将停止运算，其输出状态如下所示。

- 电源模块的过电流保护装置或者过电压保护装置动作时将所有输出置为 OFF。
- CPU 模块中通过自诊断功能检测到诸如看门狗定时器出错的异常时，根据参数设置保持或 OFF 所有输出。

此外，如果发生了可编程控制器 CPU 无法检测的 I/O 控制部分等的异常，则所有输出可能变为 ON。应在可编程控制器外部构建一个失效安全电路及安全机构以保障机械设备的安全。关于失效安全电路的示例，请参阅 MELSEC-L CPU 模块用户手册（硬件设计/维护点检篇）中的“失效安全电路的思路”。

- 应将电路配置为先启动可编程控制器的主机电源，然后投入外部供给电源。如果先启动外部供给电源，有可能导致误输出、误动作而发生事故。

## [设计注意事项]

### 警告

- 将 CPU 模块与外围设备相连接，或者将智能型功能模块与个人计算机等相连接，对运行中的可编程控制器进行控制(数据变更)时，应在顺控程序上配置互锁电路，确保整个系统始终都会安全运行。

此外，对运行中的可编程控制器进行其它控制(程序变更、运行状态变更(状态控制))时，应在熟读手册，充分确认安全后进行操作。

特别是在通过外围设备对远处的可编程控制器进行上述控制时，有时会发生由于数据通讯异常导致不能对可编程控制器端的故障进行及时处理的现象。应在顺控程序上配置互锁电路的同时，在外围设备与可编程控制器 CPU 之间确定发生数据通讯异常时的系统方面的处理方法等。

## [启动-维护保养时的注意事项]

### 注意

- 将外围设备连接在运行中的 CPU 模块上进行在线操作(特别是进行程序变更、强制输出、运行状态变更时)时，应在熟读手册，充分确认安全后进行操作。如果发生操作错误，将导致设备损坏或发生事故。

## • 关于产品的应用 •

- (1) 在使用三菱可编程控制器时，应该符合以下条件：即使在可编程控制器设备出现问题或故障时也不会导致重大事故，并且应在设备外部系统地配备能应付任何问题或故障的备用设备及失效安全功能。
- (2) 三菱可编程控制器是以一般工业用途等为对象设计和制造的通用产品。因此，三菱可编程控制器不应用于以下设备·系统等特殊用途。如果用于以下特殊用途，对于三菱可编程控制器的质量、性能、安全等所有相关责任（包括但不限于债务未履行责任、瑕疵担保责任、质量保证责任、违法行为责任、制造物责任），三菱将不负责。
  - 面向各电力公司的核电站以及其它发电厂等对公众有较大影响的用途。
  - 用于各铁路公司或公用设施目的等有特殊质量保证体系要求的用途。
  - 航空航天、医疗、铁路、焚烧·燃料装置、载人移动设备、载人运输装置、娱乐设备、安全设备等预计对人身财产有较大影响的用途。

然而，对于上述应用，如果在限于具体用途，无需特殊质量（超出一般规格的质量等）要求的条件下，经过三菱的判断也可以使用三菱可编程控制器，详细情况请与当地三菱代表机构协商。

修订记录

\*本手册号在封底的左下角。

印刷日期	* 手册编号	修改内容
1999年09月	SH(NA)-080240C-A	第一版
2006年11月	SH(NA)-080240C-B	第二版 全面改版
2008年05月	SH(NA)-080240C-C	第三版 全面改版
2011年04月	SH(NA)-080240C-D	第四版 全面改版

日文手册原稿：SH-080022-N

本手册不授予工业产权或任何其它类型的权利，也不授予任何专利许可。三菱电机对由于使用了本手册中的内容而引起的涉及工业产权的任何问题不承担责任。

© 1999 MITSUBISHI ELECTRIC CORPORATION

## 前言

非常感谢您选购了三菱通用可编程控制器的 MELSEC-Q/L/QnA 系列。  
请在使用之前熟读本书，在充分理解 Q/L/QnA 系列可编程控制器的功能、性能的基础上正确地加以使用。  
应确保将本手册交给最终用户。

安全注意事项 .....	A - 1
关于产品的应用 .....	A - 3
修订记录 .....	A - 4
前言 .....	A - 5
目录 .....	A - 5
关于手册 .....	A - 8
本手册中使用的总称和简称 .....	A - 11

## 目录

<b>1. 概述</b> .....	<b>1 - 1 到 1 - 4</b>
1.1 PID 处理方法 .....	1 - 3
<b>2. PID 控制的系统配置</b> .....	<b>2 - 1 到 2 - 4</b>
2.1 适用可编程控制 CPU .....	2 - 2
<b>3. PID 控制规格</b> .....	<b>3 - 1 到 3 - 14</b>
3.1 不完全微分 PID 控制 .....	3 - 1
3.1.1 性能规格 .....	3 - 1
3.1.2 PID 运算块图及运算公式 .....	3 - 2
3.1.3 PID 控制指令列表 .....	3 - 3
3.2 完全微分 PID 控制 .....	3 - 8
3.2.1 性能规格 .....	3 - 8
3.2.2 PID 运算块图及运算公式 .....	3 - 9
3.2.3 PID 控制指令列表 .....	3 - 10
<b>4. PID 控制的功能</b> .....	<b>4 - 1 到 4 - 12</b>
4.1 PID 控制的概述 .....	4 - 1
4.2 PID 控制的功能 .....	4 - 2
4.2.1 运算方式 .....	4 - 2
4.2.2 正向动作和逆向动作 .....	4 - 2
4.2.3 比例动作 (P 动作) .....	4 - 3
4.2.4 积分动作 (I 动作) .....	4 - 4
4.2.5 微分动作 (D 动作) .....	4 - 5
4.2.6 PID 动作 .....	4 - 7
4.3 其它功能 .....	4 - 7
4.3.1 无冲击切换功能 .....	4 - 8
4.3.2 MV 上限/下限控制功能 .....	4 - 8
4.3.3 通过 AD57 (S1) 监视 PID 控制 (仅用于 QnACPU) .....	4 - 9

4.3.4 将手动模式下的 PV 传送到 SV 存储软元件的功能.....	4 - 10
4.3.5 PID 控制数据及 I/O 设置范围的变更功能(用于 QCPU、LCPU).....	4 - 11

5. PID 控制步骤	5 - 1 到 5 - 24
-------------	----------------

5.1 PID 控制用数据.....	5 - 4
5.1.1 使用环路数和一次扫描的执行环路数.....	5 - 15
5.1.2 采样周期.....	5 - 16
5.2 I/O 数据.....	5 - 18

6. PID 控制指令	6 - 1 到 6 - 2
-------------	---------------

7. 怎样读指令说明	7 - 1 到 7 - 2
------------	---------------

8. 不完全微分 PID 控制指令和程序示例	8 - 1 到 8 - 14
------------------------	----------------

8.1 PID 控制指令.....	8 - 1
8.1.1 PID 控制用数据的设置	S. PIDINIT、SP. PIDINIT..... 8 - 2
8.1.2 PID 运算	S. PIDCONT、SP. PIDCONT..... 8 - 3
8.1.3 停止/开始指定环路号的运算	S. PIDSTOP、SP. PIDSTOP、S. PIDRUN、SP. PIDRUN... 8 - 5
8.1.4 指定环路的参数变更	S. PIDPRMW、SP. PIDPRMW..... 8 - 6
8.2 PID 控制程序示例.....	8 - 7
8.2.1 程序示例的系统配置.....	8 - 7
8.2.2 自动模式 PID 控制的程序示例.....	8 - 8
8.2.3 在自动和手动间切换 PID 控制模式的程序示例.....	8 - 12

9. 微分 PID 控制指令和程序示例	9 - 1 到 9 - 28
---------------------	----------------

9.1 PID 控制指令.....	9 - 1
9.1.1 PID 控制数据的设置	PIDINIT、PIDINITP..... 9 - 2
9.1.2 PID 控制	PIDCONT、PIDCONTP..... 9 - 3
9.1.3 监视 PID 控制状态(仅用于 QnACPU)	PID57、PID57P..... 9 - 5
9.1.4 停止/开始指定环路号的运算	PIDSTOP、PIDSTOPP、PIDRUN、PIDRUNP..... 9 - 8
9.1.5 更改指定环路号的参数	PIDPRMW、PIDPRMWP..... 9 - 9
9.2 PID 控制程序示例(用于 QCPU、LCPU).....	9 - 10
9.2.1 程序示例的系统配置.....	9 - 10
9.2.2 自动模式 PID 控制的程序示例.....	9 - 11
9.2.3 在自动和手动间切换 PID 控制模式的程序示例.....	9 - 14
9.3 PID 控制程序示例(仅用于 QnACPU).....	9 - 17
9.3.1 程序示例的系统配置.....	9 - 17
9.3.2 自动模式 PID 控制的程序示例.....	9 - 18
9.3.3 在自动和手动间切换 PID 控制模式的程序示例.....	9 - 22



附录 1 处理时间列表..... 附录 - 1  
附录 2 抗积分饱和措施..... 附录 - 5

## 关于手册

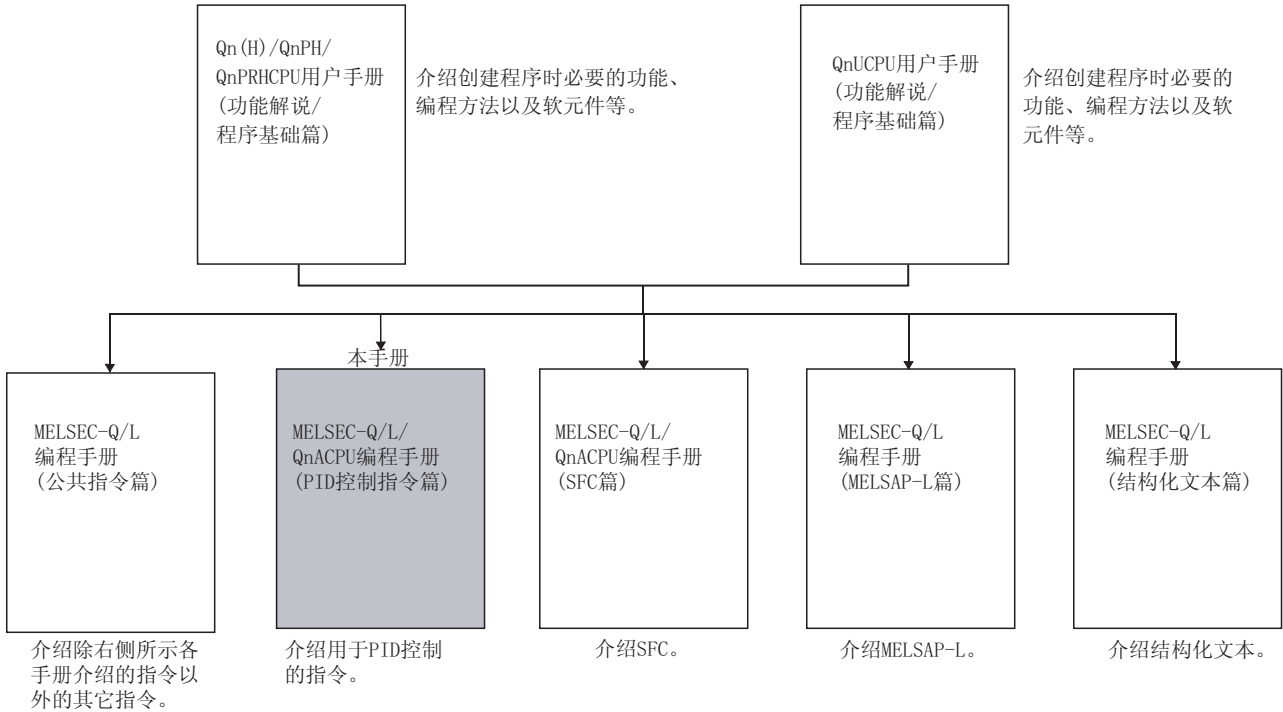
与本产品有关的手册如下表所示。  
请根据需要参考下表订购。

### 关联手册

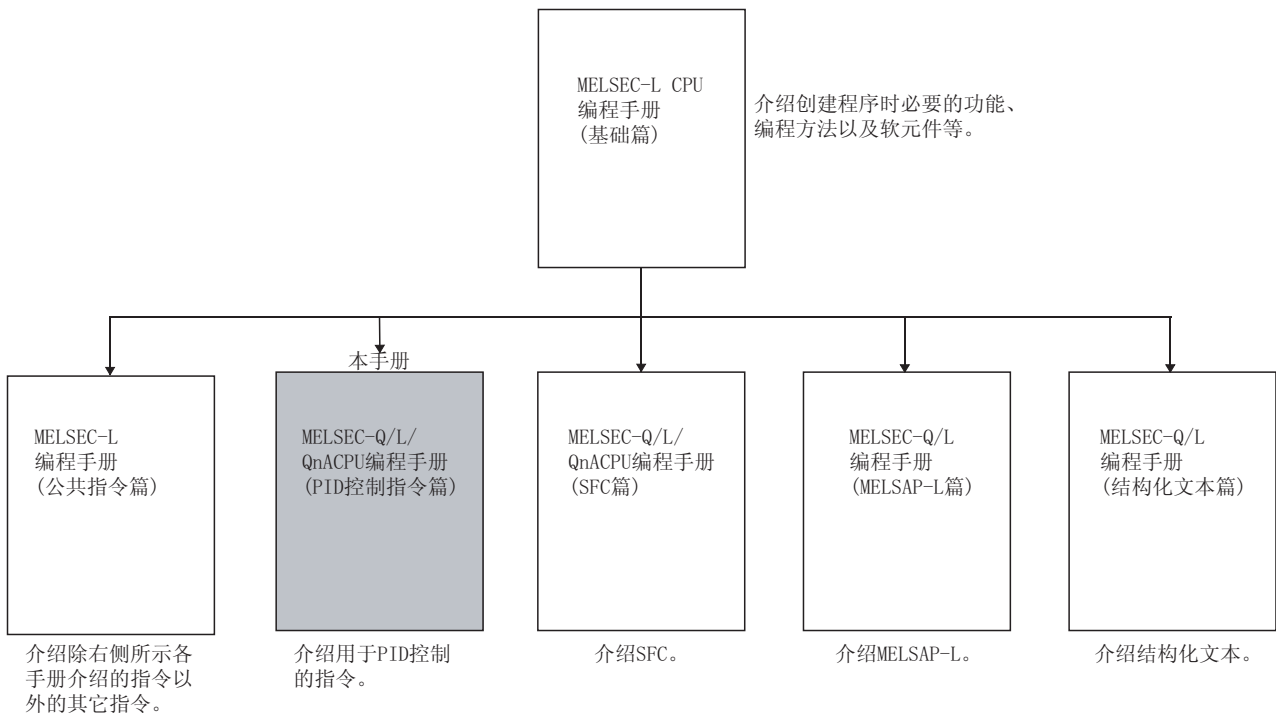
手册名称	手册编号
QCPU 用户手册(功能解说/程序基础篇) 介绍创建程序时必要的功能、编程方法、软元件等。(另售)	SH-080503CHN
Qn(H)/QnPH/QnPRHCPU 用户手册(功能解说/程序基础篇) 介绍创建程序时必要的编程方法、软元件名称、参数及程序类型等。(另售)	SH-080808ENG
MELSEC-L CPU 用户手册(功能解说/程序基础篇) 介绍创建程序时必要的功能、编程方法、软元件等。(另售)	SH-080942CHN
QnACPU 编程手册(基础篇) 介绍创建程序时必要的编程方法、软元件名称、参数及程序类型等。(另售)	SH-080229C
MELSEC-Q/L 编程手册(公共指令篇) 介绍 Q 系列的顺控程序指令、基本指令以及应用指令的使用方法有关内容。(另售)	SH-080814CHN
QnACPU 编程手册(公共指令篇) 介绍 QnA 系列的顺控程序指令、基本指令以及应用指令的使用方法有关内容。(另售)	SH-080810ENG
QnACPU 编程手册(特殊功能模块篇) 介绍 Q2ACPU(S1)、Q3ACPU、Q4ACPU 中使用的特殊功能模块用的专用指令。(另售)	SH-4013
QnACPU 编程手册(AD57 指令篇) 介绍 Q2ASCPU(S1)、Q2ASHCPU(S1)、Q2ACPU(S1)、Q3ACPU、Q4ACPU、Q4ARCPU 中用于控制 AD57(S1)型 CRT 控制器模块的专用指令。(另售)	IB-66617
MELSEC-Q/L/QnA 编程手册(SFC 篇) 介绍 MELSAP3 的系统配置、性能规格、功能、编程、调试以及出错代码等有关内容。(另售)	SH-080283CHN
MELSEC-Q/L/QnA 编程手册(MELSAP-L 篇) 介绍创建 MELSAP-L 格式的 SFC 程序时必要的编程方法、规格、功能等有关内容。(另售)	SH-080973CHN
MELSEC-Q/L 编程手册(结构化文本篇) 介绍结构化文本语言的编程方法有关内容。(另售)	SH-080907CHN

阅读本手册之前，应参阅所使用的 CPU 模块或 QnACPU 编程手册(基础篇)，预先确认在所使用的 CPU 模块中可使用的程序、I/O 处理和软元件。

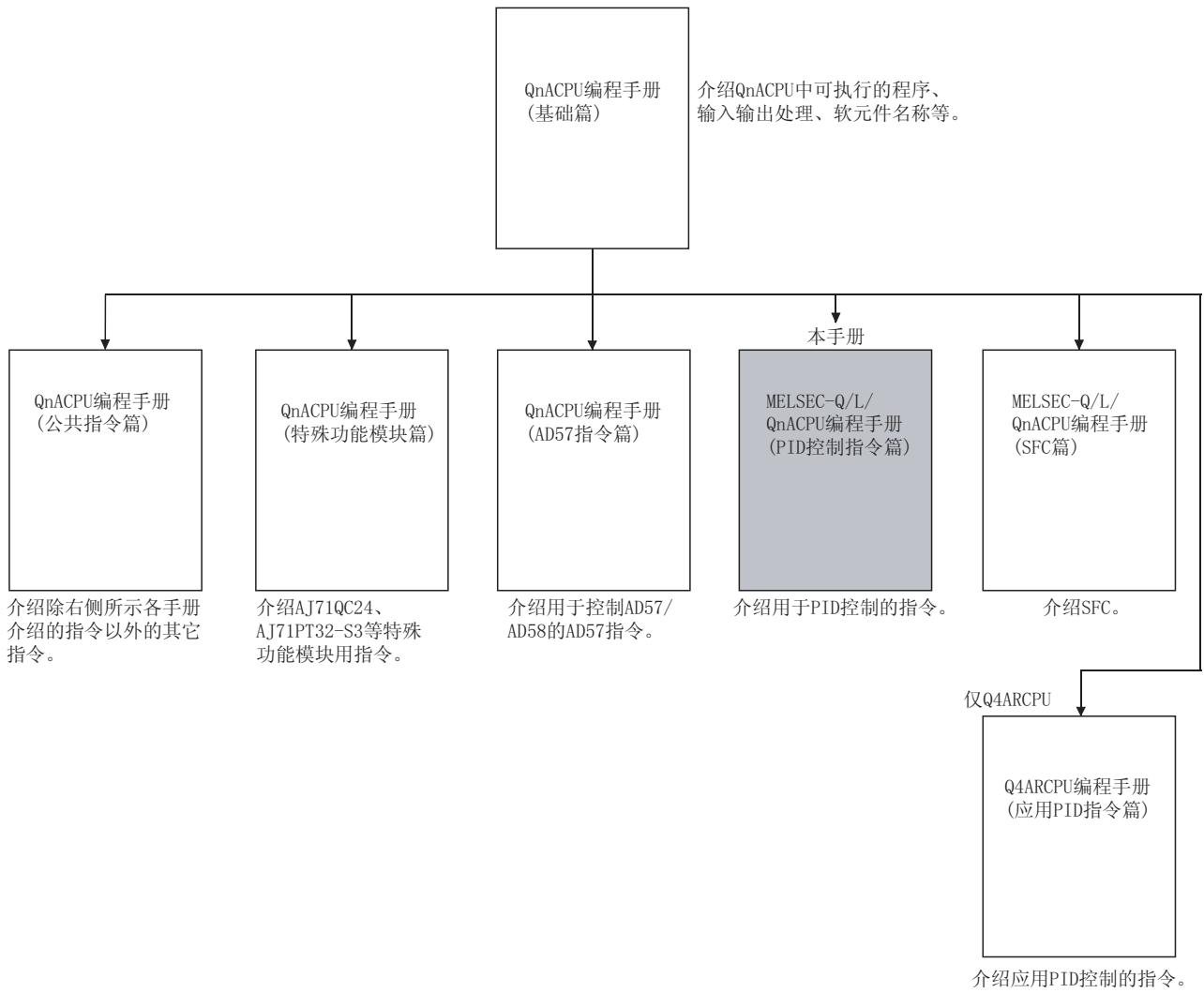
(1) 使用 QCPU 时



(2) 使用 LCPU 时



(3) 使用 QnACPU 时



## 本手册中使用的总称和简称

除非特别标明，本手册使用下列总称和简称。

总称/简称	说明
CPU 模块	基本型 QCPU、高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU、QnACPU 的总称。
QCPU	Q00CPU、Q01CPU、Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU、Q12PRHCPU、Q25PRHCPU、Q00JCPU、Q00UCPU、Q01UCPU、Q02UCPU、Q03UDCPU、Q04UDHCPU、Q06UDHCPU、Q10UDHCPU、Q13UDHCPU、Q20UDHCPU、Q26UDHCPU、Q03UDECPU、Q04UDEHCPU、Q06UDEHCPU、Q10UDEHCPU、Q13UDEHCPU、Q20UDEHCPU、Q26UDEHCPU、Q50UDEHCPU、Q100UDEHCPU 的总称。
QnCPU	Q02CPU 的总称。
QnHCPU	Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU 的总称。
QnPHCPU	Q12PHCPU、Q25PHCPU 的总称。
QnPRHCPU	Q12PRHCPU、Q25PRHCPU 的总称。
LCPU	L02CPU、L26CPU-BT 的总称。
QnACPU	Q2ASCPU、Q2ASCPU-S1、Q2ASHCPU、Q2ASHCPU-S1、Q2ACPU、Q2ACPU-S1、Q3ACPU、Q4ACPU、Q4ARCPU 的简称。
QnA	Q2ASCPU、Q2ASCPU-S1、Q2ASHCPU、Q2ASHCPU-S1、Q2ACPU、Q2ACPU-S1、Q3ACPU、Q4ACPU 的总称。
Q4AR	Q4ARCPU 的总称。
基本型 QCPU	Q00JCPU、Q00CPU、Q01CPU 的总称。
基本	
高性能型 QCPU	Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU 的总称。
高性能	
过程 CPU	Q02PHCPU、Q06PHCPU、Q12PHCPU、Q25PHCPU 的总称。
冗余 CPU	Q12PRHCPU、Q25PRHCPU 的总称。
通用型 QCPU	Q00JCPU、Q00UCPU、Q01UCPU、Q02UCPU、Q03UDCPU、Q04UDHCPU、Q06UDHCPU、Q10UDHCPU、Q13UDHCPU、Q20UDHCPU、Q26UDHCPU、Q03UDECPU、Q04UDEHCPU、Q06UDEHCPU、Q10UDEHCPU、Q13UDEHCPU、Q20UDEHCPU、Q26UDEHCPU、Q50UDEHCPU、Q100UDEHCPU 的总称。
通用	

# 备忘录

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

## 1. 概述

本手册介绍下列 CPU 模块中用于执行 PID 控制的顺控程序指令。

- 基本型 QCPU (序列号的前五位为 04122 或以后)
- 高性能型 QCPU
- 冗余 CPU
- 通用型 QCPU
- LCPU
- QnACPU

基本型 QCPU、高性能型 QCPU、冗余 CPU、通用型 QCPU 以及 LCPU 标准配备了用于执行不完全微分 PID 控制的指令 (PID 控制指令) 和用于执行完全微分 PID 控制的指令 (PID 控制指令)。

QnACPU 标准配备了用于执行完全微分 PID 控制的指令 (PID 控制指令)。

因为不完全微分 PID 控制指令和完全微分 PID 控制指令是相互独立的，因此可以同时执行。

下表给出了可以使用不完全微分 PID 控制指令和完全微分 PID 控制指令的 CPU 模块。

CPU 模块型号		不完全微分	完全微分
基本型 QCPU	序列号的前 5 位为“04121”或以前	×	×
	序列号的前 5 位为“04122”或以后	○	○
高性能型 QCPU	序列号的前 5 位为“05031”或以前	×	○
	序列号的前 5 位为“05032”或以后	○*1	○
冗余 CPU		○	○
通用型 QCPU		○	○
LCPU		○	○
QnACPU		×	○

○: 可使用; ×: 不可使用

\*1: 如果使用版本 7 或以前版本的 GX Developer 读取 GX Developer 版本 8 中安装的 CPU 指令，GX Developer 会将其作为“指令代码异常”处理。

以下为 PID 控制指令。

分类	不完全微分	完全微分
PID 控制数据设置	S(P).PIDINIT	PIDINIT(P)
PID 运算	S(P).PIDCONT	PIDCONT(P)
PID 控制状态监视	—	PID57(P)
指定环路的运算停止	S(P).PIDSTOP	PIDSTOP(P)
指定环路的运算开始	S(P).PIDRUN	PIDRUN(P)
指定环路的参数变更	S(P).PIDPRMW	PIDPRMW(P)

PID 控制指令将 CPU 模块与 A/D 转换模块和 D/A 转换模块组合后执行 PID 控制。在 QnACPU 中，可以使用 AD57(S1) 型 CRT 控制模块监视 PID 控制状态。

要点
----

- |   |
|---|
| <p>(1) 过程 CPU 不支持本手册中介绍的 PID 控制指令。<br/>若想使用过程 CPU 执行 PID 控制，应使用 QnPHCPU/QnPRHCPU 编程手册(过程控制指令篇)中介绍的过程控制指令。</p> <p>(2) 冗余 CPU 可以使用 PID 控制指令和过程控制指令。</p> |
|---|



# 1. 概述

## 1.1 PID 处理方法

本节介绍使用 PID 控制指令进行 PID 控制的处理方法。(关于 PID 运算的详细内容, 请参阅第 4 章)

通过与 A/D 转换模块和 D/A 转换模块的组合, 使用 PID 控制指令执行 PID 控制, 如图 1.1 所示。

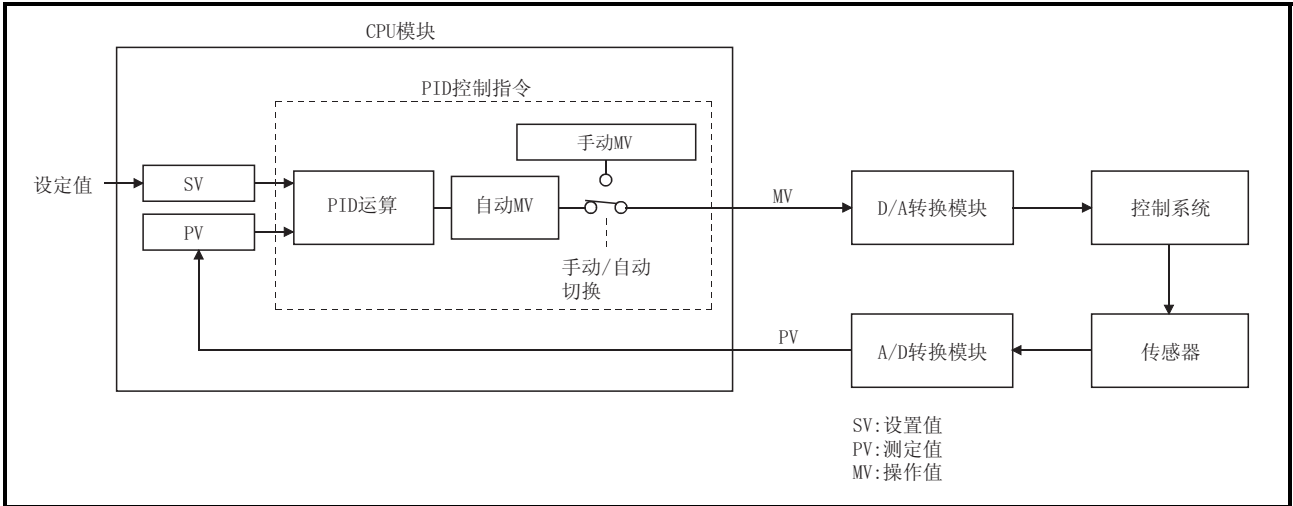


图 1.1 PID 控制处理概述

在 PID 控制处理方法中, 如图 1.1 所示, 通过预先设置的设置值 (SV) 和从 A/D 转换模块中读取的测定值 (PV) 计算出执行 PID 运算的操作值 (MV)。将计算出的操作值 (MV) 写入 D/A 转换模块后输出到外部。

在顺控程序中执行 PID 运算指令\*时, 测定采样周期并执行 PID 运算。在预设的各个采样周期中执行 PID 运算指令的 PID 运算。

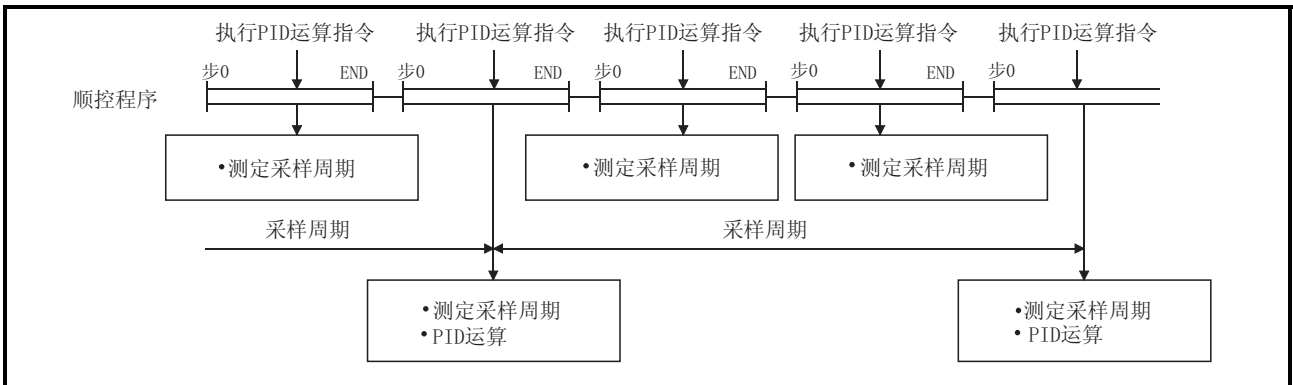


图 1.2 执行 PID 运算指令时的动作

### 备注

- \*: 下列为 PID 运算指令
  - S. PIDCONT (不完全微分)
  - PIDCONT (完全微分)

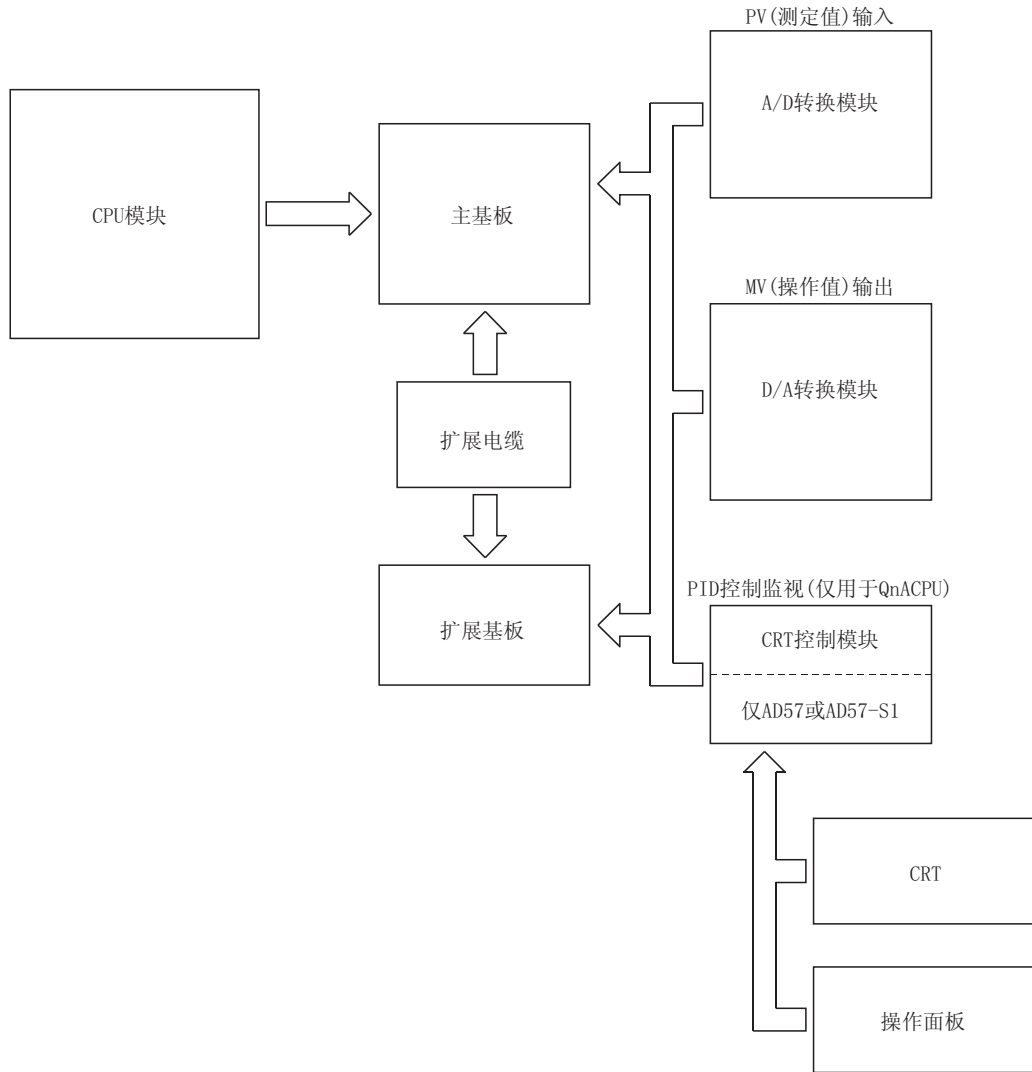


## 2. PID 控制的系统配置

本章介绍使用 PID 控制指令的 PID 控制的系统配置。

关于配置 PID 控制系统时可使用的模块，请参阅下列手册。

- 基本型 QCPU、高性能型 QCPU、通用型 QCPU：MELSEC-Q 数据手册
- QnACPU：使用的 CPU 模块的用户手册(详细篇)。



要点		
对于 PID 控制指令中使用的 SV、PV 和 MV，可以设置成 0 至 2000 的固定值或根据使用的模块设置成任意值。 相关细节请参阅 4.3.5 项。		
CPU 模块型号	SV、PV、MV	
	0 至 2000 固定值*	设置任意值
基本型 QCPU	○	○
高性能型 QCPU	○	○
冗余 CPU	○	○
通用型 QCPU	○	○
LCPU	○	○
QnACPU	○	×

○: 可设置, ×: 不可设置

\*: 当用于 PID 控制的 I/O 的 A/D 转换模块或 D/A 转换模块的分辨率不在 0 至 2000 之间时, 应将数字值转换为 0 至 2000 之间。

## 2. PID 控制的系统配置

### 2.1 适用可编程控制 CPU

产品名称	型号
基本型 QCPU	Q00JCPU、Q00CPU、Q01CPU (序列号的前 5 位为 04122 或以后)
高性能型 QCPU	Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU
冗余 CPU	Q12PRHCPU、Q25PRHCPU
通用型 QCPU	Q00UJCPU、Q00UCPU、Q01UCPU、Q02UCPU、Q03UDCPU、 Q04UDHCPU、Q06UDHCPU、Q10UDHCPU、Q13UDHCPU、 Q20UDHCPU、Q26UDHCPU、Q03UDECPU、Q04UDEHCPU、 Q06UDEHCPU、Q10UDEHCPU、Q13UDEHCPU、 Q20UDEHCPU、Q26UDEHCPU、Q50UDEHCPU、 Q100UDEHCPU
LCPU	L02CPU、L26CPU-BT
QnACPU	Q2ASCPU、Q2ASCPU-S1、Q2ASHCPU、Q2ASHCPU-S1、Q2ACPU、 Q3ACPU、Q4ACPU、Q4ARCPU



### 3. PID 控制规格

### 3. PID 控制规格

本节介绍 PID 控制指令中的 PID 运算规格。

#### 3.1 不完全微分 PID 控制

##### 3.1.1 性能规格

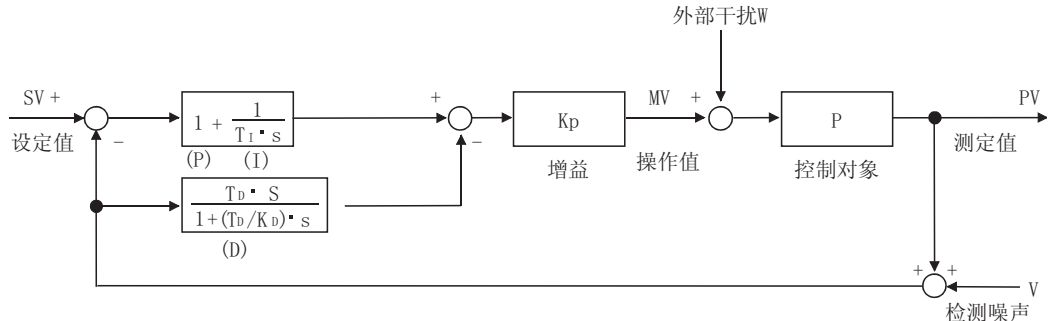
下表为 PID 控制性能规格。

项目		规格				
		有 PID 上下限制		无 PID 上下限制		QnA CPU
		基本型 QCPU	高性能型 QCPU、 冗余 CPU、 通用型 QCPU、 LCPU	基本型 QCPU	高性能型 QCPU、 通用型 QCPU、 LCPU	
PID 控制环路数量	—	8 个环路 (最多)	32 个环路 (最多)	8 个环路 (最多)	32 个环路 (最多)	—
采样周期	T <sub>s</sub>	0.01 至 60.00s				—
PID 运算方式	—	测定值微分型不完全微分(正向动作/逆向动作)				—
PID 常数 设置范围	比例常数	K <sub>P</sub>	0.01 至 100.00			—
	积分常数	T <sub>I</sub>	0.1 至 3000.0s			—
	微分常数	T <sub>D</sub>	0.00 至 300.00s			—
	微分增益	K <sub>D</sub>	0.00 至 300.00			—
SV(设置值)设置范围	SV	0 至 2000		-32768 至 32767		—
PV(测定值)设置范围	PV	-50 至 2050		-32768 至 32767		—
MV(操作值)输出范围	MV					

—: 不可使用

3.1.2 PID 运算块图及运算公式

(1) 不完全微分的 PID 运算块图所示如下所示：



(2) 下表为使用 PID 控制指令进行 PID 运算时的运算公式。

名称	运算表达式	符号意义
测定值微分型不完全微分	正向动作 $EV_n = PV_{fn}^* - SV$ $\Delta MV = K_p \left\{ (EV_n - EV_{n-1}) + \frac{T_s}{T_i} EV_n + D_n \right\}$ $D_n = \frac{T_d}{T_s + \frac{T_d}{K_d}} (PV_{fn} - 2PV_{fn-1} + PV_{fn-2}) + \frac{\frac{T_d}{K_d}}{T_s + \frac{T_d}{K_d}} D_{n-1}$ $MV_n = \sum \Delta MV$	$EV_n$ : 在当前采样周期中的偏差 $EV_{n-1}$ : 在上一个采样周期中的偏差 $SV$ : 设置值 $PV_{fn}$ : 当前采样时的测定值(过滤后) $PV_{fn-1}$ : 上一个采样周期的测定值(过滤后) $PV_{fn-2}$ : 两个周期前采样周期的测定值(过滤后)
	逆向动作 $EV_n = SV - PV_{fn}^*$ $\Delta MV = K_p \left\{ (EV_n - EV_{n-1}) + \frac{T_s}{T_i} EV_n + D_n \right\}$ $D_n = \frac{T_d}{T_s + \frac{T_d}{K_d}} (-PV_{fn} + 2PV_{fn-1} - PV_{fn-2}) + \frac{\frac{T_d}{K_d}}{T_s + \frac{T_d}{K_d}} D_{n-1}$ $MV_n = \sum \Delta MV$	$\Delta MV$ : 输出变化值 $MV_n$ : 当前操作值 $D_n$ : 当前微分项 $D_{n-1}$ : 上一个采样周期的微分项 $K_p$ : 比例常数 $T_s$ : 采样周期 $T_i$ : 积分常数 $T_d$ : 微分常数 $K_d$ : 微分增益

要点

(1) \*:  $PV_{fn}$  为使用下列公式计算的输入数据的测定值。因此，未对输入数据设置过滤器系数时， $PV_{fn}$  与输入数据的测定值(PV)相同。

$$\text{过滤后的测定值 } PV_{fn} = PV_n + \alpha (PV_{fn-1} - PV_n)$$

$PV_n$  : 当前采样周期的测定值

$\alpha$  : 过滤器系数

$PV_{fn-1}$  : 上一个采样周期的测定值(过滤后)

(2)  $PV_{fn}$  存储于 I/O 数据区。(参阅 5.2 节)



### 3. PID 控制规格

#### 3.1.3 PID 控制指令列表

下表列出了可用于执行 PID 控制的指令。

指令名称	处理内容	对象 CPU	
		QCPU、LCPU	QnACPU
S. PIDINIT	为 PID 运算设置作为参照基准的数据。	○*	×
S. PIDCONT	根据所设置的 SV(设置值)和 PV(测定值)执行 PID 运算。	○*	×
S. PIDSTOP S. PIDRUN	停止或开始指定环路的 PID 运算。	○	×
S. PIDPRMW	将指定环路的运算参数更改成 PID 控制数据。	○*	×

○：可使用，×：不可使用




\*：在基本型 QCPU、高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU 中允许选择“有/无 PID 上下限的限制”。

有关选择了“有/无 PID 上下限的限制”时的设置范围的详细内容，请参阅 5.1 和 5.2 节。

(1) PID 控制指令列表

PID 控制指令列表有如下所示的格式。

表 3.1 PID 控制指令列表的阅读方法

分类	指令符号	梯形图格式	处理内容	执行条件	基本步数	子集	页码
PID控制数据设置	S. PIDINIT	 	设置Ⓢ中指定的字软元件中存储的PID控制数据。 Ⓢ+0 公共数据设置区 Ⓢ+1 至 Ⓢ+15 环路1用 Ⓢ+16 至 Ⓢ+29 环路2用 至 环路n用 Ⓢ+(m+0) 至 Ⓢ+(m+13) $m=(n-1) \times 14+2$		7	—	8-2

(1) (2) (3) (4) (5) (6) (7) (8)

说明

- (1) 根据不同用途将指令分类。
- (2) 顺控程序中使用的指令代码。
- (3) 用于梯形图中的符号。
- (4) 各指令的处理内容。

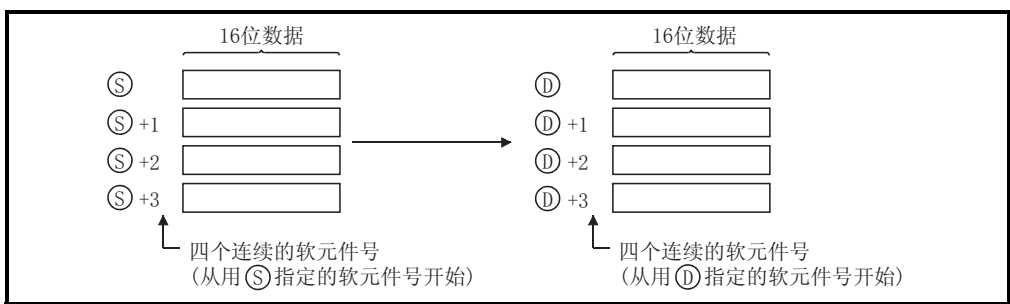




图 3.1 各指令的处理内容

(5) 各指令的执行条件的详细内容如下所示。

符号	执行条件
	只在 ON 中执行的指令。只有该指令的前条件为 ON 期间才执行该指令。 前条件为 OFF 时，不执行该指令且不进行处理。
	是 ON 时一次执行型指令。仅在该指令的前条件从 OFF 变为 ON 的上升沿时执行一次该指令。以后即使条件为 ON，也不执行该指令且不进行处理。

(6) 指令的步数

有关步数的详细内容，请参阅 QCPU(Q 模式)/QnACPU 编程手册(公共指令篇)。

(7) ○表示可以进行子集处理。

-表示不可以进行子集处理。

有关子集处理的详细内容，请参阅所使用的 CPU 模块的编程手册(公共指令篇)。

(8) 表示本手册中有关指令的详细介绍的页码。

### 3. PID 控制规格

表 3.2 给出了 PID 控制指令列表。

表 3.2 PID 控制指令列表

分类	指令符号	梯形图格式	处理内容	执行条件	基本步数	子集	页码
PID 控制数据设置	S. PIDINIT		设置存储于字软元件 (由Ⓢ指定) 的 PID 控制数据。 Ⓢ+0 公共数据设置区 Ⓢ+1 至 Ⓢ+15 环路1用 Ⓢ+16 至 Ⓢ+29 环路2用 至 Ⓢ+(m+0) 至 Ⓢ+(m+13) 环路n用 $m=(n-1) \times 14+2$		7	-	8-2
PID 运算	S. PIDCONT		用由Ⓢ指定的 SV (设置值) 和 PV (测定值) 执行 PID 运算, 然后将结果存储于由Ⓢ指定的字软元件的 MV (操作值) 区。 Ⓢ 至 Ⓢ+9 公共数据设置区 Ⓢ+10 至 Ⓢ+31 SV 值设置区 PV 值设置区 MV 值存储区 至 Ⓢ+32 至 Ⓢ+54 SV 值设置区 PV 值设置区 MV 值存储区 Ⓢ+55 至 Ⓢ+(m+0) 至 Ⓢ+(m+21) SV 值设置区 PV 值设置区 MV 值存储区 Ⓢ+(m+22) 至 $m=(n-1) \times 23+10$		7	-	8-3
停止运算	S. PIDSTOP		停止在Ⓢ中指定的环路号的 PID 运算。		7	-	8-5
开始运算	S. PIDRUN		开始在Ⓢ中指定的环路号的 PID 运算。		6	-	8-5
变更参数	S. PIDPRMW		将Ⓢ中指定的环路号的运算参数更改为Ⓢ中指定的字软元件中存储的 PID 控制数据。		8	-	8-6

要点
----

- |  |
|--|
| <p>(1) “不完全微分 PID 运算”和“完全微分 PID 运算”是相互独立的，因此可以同时执行。</p> <p>(2) 通过 S(P).PIDINIT 指令进行了初始化时，应使用 S(P).PIDCONT 指令执行 PID 运算。</p> <p>若想停止和开始指定环路号的 PID 运算及变更 PID 控制数据，应使用 S(P).PIDSTOP、S(P).PIDRUN 及 S(P).PIDPRMW 指令。</p> |
|--|

### 3. PID 控制规格

#### 3.2 完全微分 PID 控制

##### 3.2.1 性能规格

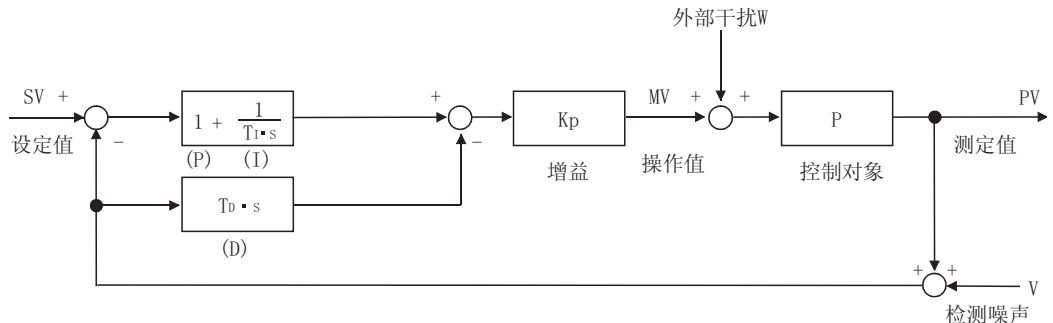
PID 控制的性能规格如下表所示。

项目		规格				
		有 PID 上下限限制		无 PID 上下限限制		QnACPU
		基本型 QCPU	高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU	基本型 QCPU	高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU	
PID 控制环路数量	—	8 个环路 (最多)	32 个环路 (最多)	8 个环路 (最多)	32 个环路 (最多)	32 个环路 (最多)
采样周期	$T_s$	0.01 至 60.00s				
PID 运算方式	—	测定值微分型完全微分(正向动作/逆向动作)				
PID 常数 设置范围	比例常数	$K_P$	0.01 至 100.00			
	积分常数	$T_I$	0.1 至 3000.0s			
	微分常数	$T_D$	0.00 至 300.00s			
SV(设置值)设置范围	SV	0 至 2000		-32768 至 32767		0 至 2000
PV(测定值)设置范围	PV					
MV(操作值)输出范围	MV		-50 至 2050		-32768 至 32767	-50 至 2050

### 3. PID 控制规格

#### 3.2.2 PID 运算块图及运算公式

(1) 完全微分的 PID 运算块图如下所示：



(2) 使用 PID 控制指令的 PID 运算的运算公式如下所示。

名称	运算表达式	符号意义
测定值微分型	正向动作 $EV_n = PV_{fn}^* - SV$ $\Delta MV = K_P \left\{ (EV_n - EV_{n-1}) + \frac{T_S}{T_I} EV_n + D_n \right\}$ $D_n = \frac{T_D}{T_S} \cdot (PV_{fn} - 2PV_{fn-1} + PV_{fn-2})$ $MV_n = \sum \Delta MV$	$EV_n$ : 在当前采样周期中的偏差 $EV_{n-1}$ : 在上一个采样周期中的偏差 $SV$ : 设置值 $PV_{fn}$ : 当前采样周期的测定值(过滤后) $PV_{fn-1}$ : 上一个采样周期的测定值(过滤后) $PV_{fn-2}$ : 两个周期之前采样周期的测定值(过滤后) $\Delta MV$ : 输出变化值
	完全微分 逆向动作 $EV_n = SV - PV_{fn}^*$ $\Delta MV = K_P \left\{ (EV_n - EV_{n-1}) + \frac{T_S}{T_I} EV_n + D_n \right\}$ $D_n = \frac{T_D}{T_S} (-PV_{fn} + 2PV_{fn-1} - PV_{fn-2})$ $MV_n = \sum \Delta MV$	$MV_n$ : 当前操作值 $D_n$ : 当前微分项 $K_P$ : 比例常数 $T_S$ : 采样周期 $T_I$ : 积分常数 $T_D$ : 微分常数

#### 要点

- (1) \*: 用下列的公式计算输入数据的测定值后得到  $PV_{fn}$ 。  
 因此，未对输入数据设置过滤器系数时， $PV_{fn}$  与输入数据的 PV(测定值) 相同。

$$\text{过滤后的测定值 } PV_{fn} = PV_n + \alpha (PV_{fn-1} - PV_n)$$

$PV_n$  : 当前采样周期的测定值

$\alpha$  : 过滤器系数

$PV_{fn-1}$  : 上一个采样周期的测定值(过滤后)

- (2)  $PV_{fn}$  存储于 I/O 数据区。(参阅 5.2 节)

### 3. PID 控制规格

#### 3.2.3 PID 控制指令列表

用于执行 PID 控制的指令列表如下所示。

指令名称	处理内容	CPU	
		QCPU、LCPU	QnACPU
PIDINIT	为 PID 运算设置作为参照基准的数据。	○*	○
PIDCONT	根据所设置的 SV (设置值) 和 PV (测定值) 执行 PID 运算。	○*	○
PID57	使用 AD57 (S1) 监视 PID 运算结果。	×	○
PIDSTOP PIDRUN	停止或开始指定环路号的 PID 运算。	○	○
PIDPRMW	将指定环路号的运算参数更改为 PID 控制数据。	○*	○

○：可使用，×：不可使用

\*：在基本型 QCPU、高性能型 QCPU、冗余 CPU、通用型 QCPU 和 LCPU 中允许选择“有/无 PID 上下限限制”。




有关选择了“有/无 PID 上下限限制”时的设置范围的详细内容，请参阅 5.1 节和 5.2 节。



(1) PID 控制指令列表

PID 控制指令列表具有的格式如下所示。

表 3.3 PID 控制指令列表的阅读方法

分类	指令符号	梯形图格式	处理内容	执行条件	基本步数	子集	页码
PID 控制数据设置	PIDINIT	 	设置存储于字软元件 (由 S 指定的)PID 控制数据  S+0 公共数据设置区 S+1 S+2 至 S+11 环路1用 S+12 至 S+21 环路2用 至 环路n用 S+(m+0) 至 S+(m+9) $m=(n-1) \times 10+2$		2	—	9-2

(1) (2) (3) (4) (5) (6) (7) (8)

说明

- (1) 按不同用途将指令分类。
- (2) 表示程序中使用的指令代码。
- (3) 用于梯形图中的符号。
- (4) 各指令的处理内容。

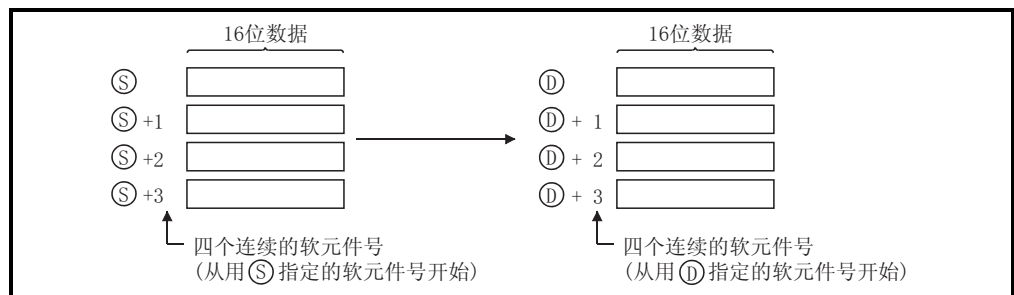




图 3.2 各指令的处理内容

(5) 各指令执行条件的详细内容如下所示。

符号	执行条件
	只在 ON 中执行的指令。只有该指令的前条件为 ON 期间才执行该指令。 前条件为 OFF 时，不执行该指令且不进行处理。
	是 ON 时一次执行型指令。仅在指令的前条件为 OFF 至 ON 的上升沿时执行一次指令。以后即使条件为 ON，也不执行指令且不进行处理。

(6) 指令的基本步数

有关步数的详细内容，请参阅所使用的 CPU 模块的编程手册(公共指令篇)。

(7) ○表示可以进行子集处理。

-表示不可以进行子集处理。

有关子集处理的详细内容，请参阅所使用的 CPU 模块的编程手册(公共指令篇)。

(8) 表示本手册中有关指令的详细介绍的页码。

### 3. PID 控制规格

表 3.4 给出了 PID 控制指令列表。

表 3.4 PID 控制指令列表







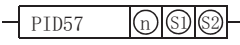


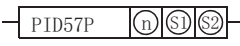
分类	指令符号	梯形图格式	处理内容	执行条件	基本步数	子集	页码
PID 控制数据设置	PIDINIT		设置存储于字软元件(用Ⓢ指定)的 PID 控制数据。 Ⓢ+0 公共数据设置区 Ⓢ+1 至 Ⓢ+11 环路1用 Ⓢ+12 至 Ⓢ+21 环路2用 至 Ⓢ+(m+0) 至 Ⓢ+(m+9) 环路n用 $m=(n-1) \times 10+2$	 	2	-	9-2
PID 运算	PIDCONT		用Ⓢ中指定的 SV(设置值)和 PV(测定值)执行 PID 运算, 然后将结果存储于Ⓢ指定的字软元件的 MV(操作值)区。 Ⓢ+0 至 Ⓢ+9 公共数据设置区 Ⓢ+10 至 Ⓢ+27 SV值设置区 PV值设置区 MV值存储区 环路1用 Ⓢ+28 至 Ⓢ+45 SV值设置区 PV值设置区 MV值存储区 环路2用 Ⓢ+(m+0) 至 Ⓢ+(m+17) SV值设置区 PV值设置区 MV值存储区 环路n用 $m=(n-1) \times 18+10$	 	2	-	9-3
监视	PID57		监视 AD57(S1)(Ⓢ中指定)的 PID 运算结果。 Ⓢ: AD57(S1)的起始 I/O 号 Ⓢ1: 监视画面号 1: 环路 1 至环路 8 2: 环路 9 至环路 16 3: 环路 17 至环路 24 4: 环路 25 至环路 32 Ⓢ2: 初始画面显示请求	 	4	-	9-5
	PID57P						

表 3.4 PID 控制指令列表

分类	指令符号	梯形图格式	处理内容	执行条件	基本步数	子集	页码
停止运算	PIDSTOP		停止Ⓜ中指定的环路号的 PID 运算。		2	-	9-8
开始运算	PIDRUN		开始Ⓜ中指定的环路号的 PID 运算。		2	-	9-8
变更参数	PIDPRMW		将Ⓜ中指定的环路号的运算参数改为Ⓜ中指定的字软元件中存储的 PID 控制数据。		3	-	9-9

要点
<p>(1) “不完全微分 PID 运算”和“完全微分 PID 运算”是相互独立的，因此可以同时执行。</p> <p>(2) 通过 PIDINIT(P) 指令进行初始化时，应使用 PIDCONT(P) 指令执行 PID 运算。若想停止和开始指定环路号的 PID 运算及变更 PID 控制数据，应使用 PIDSTOP(P) 指令、PIDRUN(P) 指令和 PIDPRMW(P) 指令。</p>

## 4. PID 控制的功能

### 4. PID 控制的功能

本章介绍使用 PID 控制指令执行 PID 控制有关内容。

#### 4.1 PID 控制的概述

PID 控制是应用于流量、速度、风量、温度、压力、配比等的过程控制的控制方式。将控制对象保持为所设置的值的所需配置如下图所示。

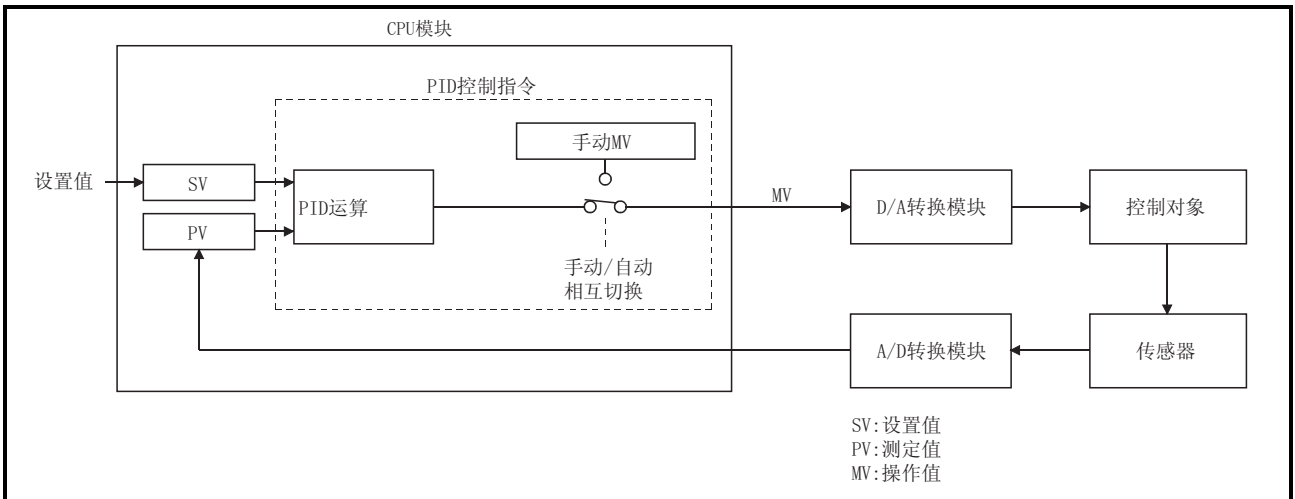


图 4.1 PID 过程控制的应用示例

PID 控制期间，将由传感器测量的值(测定值)与预先设置的值(设置值)进行比较，然后调节输出值(操作值)以消除测定值和设置值之间的差。  
在 PID 控制运算中，通过组合比例动作(P)、积分动作(I)和微分动作(D)计算 MV(操作值)，使测量值迅速、正确地趋近于设置值(SV)。  
当 PV 与 SV 的差增大时 MV 增大，迅速地使 PV 值趋近于设置值；PV 与 SV 的差变小后，减小 MV，平缓、正确地将 PV 调节为与 SV 相同。

### 4.2 PID 控制的功能

用 PID 控制指令执行 PID 控制的运算方式有速度型和测定值微分型两种。以下介绍用这两种运算方法执行的控制。

#### 4.2.1 运算方式

##### (1) 速度型运算

速度型运算是在 PID 运算中计算各个 MV (操作值) 的变化量的运算方式。实际的 MV 是各个采样周期计算出的 MV 变化量的累计值。

##### (2) 测定值微分型运算

测定值微分型运算是在 PID 运算中将 PV (测定值) 作为微分项使用的运算方式。因为在微分项中不使用偏差, 改变设置值导致偏差变化时, 由于微分动作可以减轻输出的骤变。

#### 4.2.2 正向动作和逆向动作

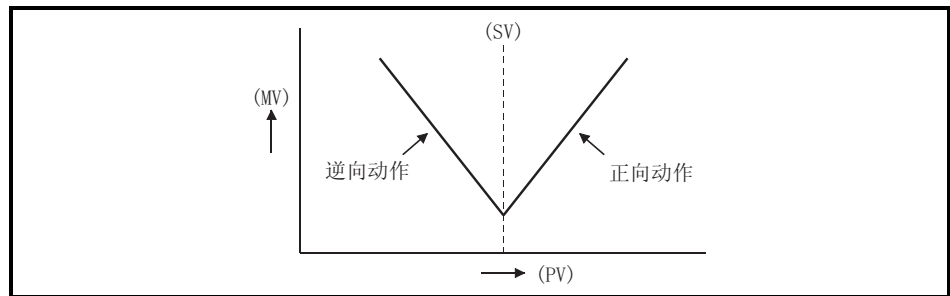
在 PID 控制中, 可以选择正向动作或逆向动作指定控制方向。

(1) 在正向动作中, 与 SV (设置值) 相比, PV (测定值) 增加时使 MV (操作值) 增加。

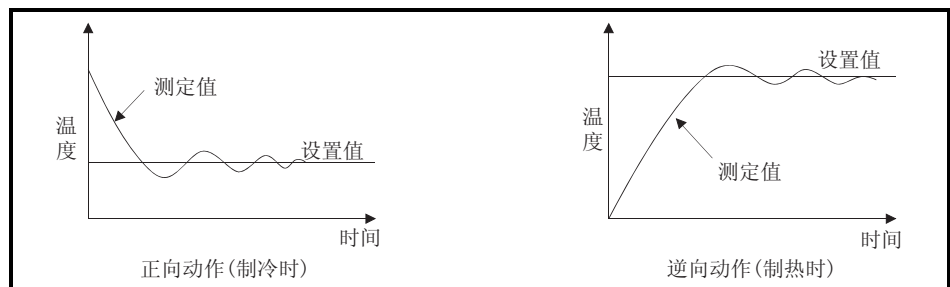
(2) 在逆向动作中, 与 SV (设置值) 相比, PV (测定值) 减小时使 MV (操作值) 增加。

(3) 无论在正向动作还是在逆向动作中, MV 都将随着 SV 与 PV 之差的增大而增大。

(4) 下图说明了正向动作和逆向动作与 MV、PV 和 SV 之间的关系。



(5) 下图给出了用正向动作和逆向动作进行过程控制的示例。



4.2.3 比例动作(P 动作)

以下介绍比例动作的控制方法。

- (1) 在比例动作中，MV(操作值)与偏差(设置值与测定值之差)成比例关系。
- (2) 下面的公式表达了 E(偏差)与 MV 之间的关系。

$$MV = K_p \cdot E$$

$K_p$  是比例常数也称为“比例增益”。

条件	比例动作
当比例增益 $K_p$ 减小时	控制动作变慢。
当比例增益 $K_p$ 增大时	控制动作变快。 但是，易发生振荡。

- (3) 恒定偏差的阶跃响应时的比例动作如图 4.2 所示。

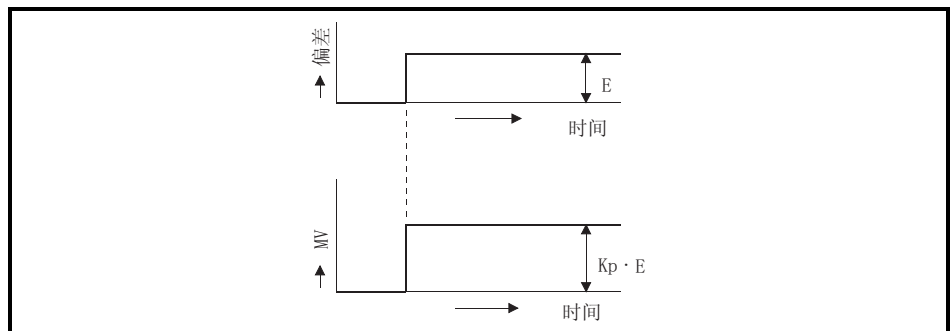
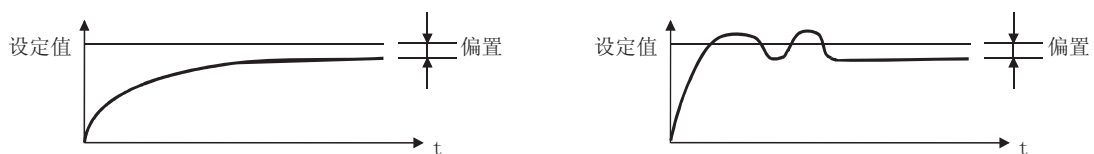


图 4.2 恒定偏差的比例动作

- (4) 产生的相对于设置值的恒定误差称为偏置(残留偏差)。  
偏置产生于比例动作中。



4.2.4 积分动作(I 动作)

以下介绍积分动作的控制方法。

- (1) 积分动作是指，存在有偏差时，连续地变化 MV（操作值）以消除偏差的动作。该动作可消除比例动作中产生的偏置。
- (2) 偏差产生后，积分动作的 MV 达到比例动作 MV 所需的时间称为积分时间。积分时间以  $T_I$  表示。

条件	积分动作
积分时间 $T_I$ 较小时	积分效果增大且消除偏置所用时间变短。但是，易发生振荡。
积分时间 $T_I$ 较大时	积分效果减小且消除偏置所用时间变长。

- (3) 偏差为恒定值的阶跃响应时的积分动作如图 4.3 所示。

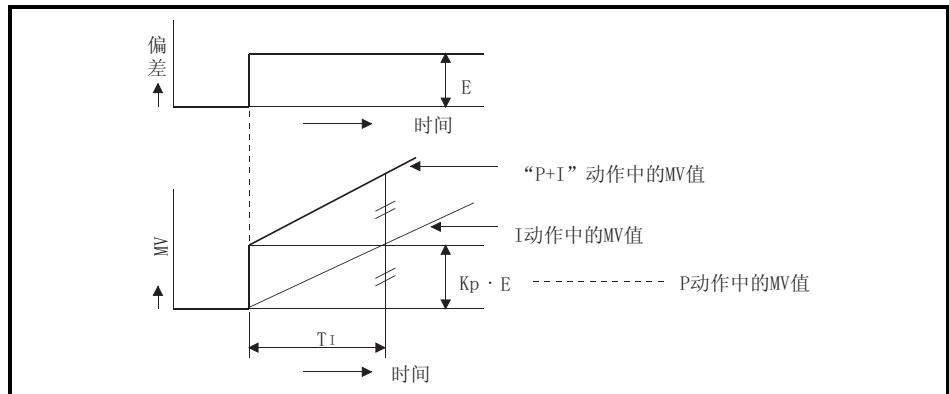


图 4.3 恒定偏差的积分动作

- (4) 积分动作常与比例动作组合使用 (PI 动作) 或与比例及微分动作组合使用 (PID 动作)。积分动作不能独立使用。



### 4.2.5 微分动作(D 动作)

以下介绍微分动作的控制方法。

- (1) 微分动作是指，产生偏差时，将与偏差随时间的变化率成正比的 MV (操作值) 施加到偏差中以消除偏差的动作。  
本动作可防止由外部干扰等导致控制目标发生大的波动。
- (2) 偏差产生后，微分动作的 MV 达到比例动作的 MV 所需的时间称为微分时间。微分时间以  $T_D$  表示。

条件	微分动作
当微分时间 $T_D$ 变短时	微分效果减小。
当微分时间 $T_D$ 变长时	微分效果增大。 但是，易产生短周期振荡。

- (3) 偏差为恒定值的阶跃响应时的微分动作如图 4.4 所示。

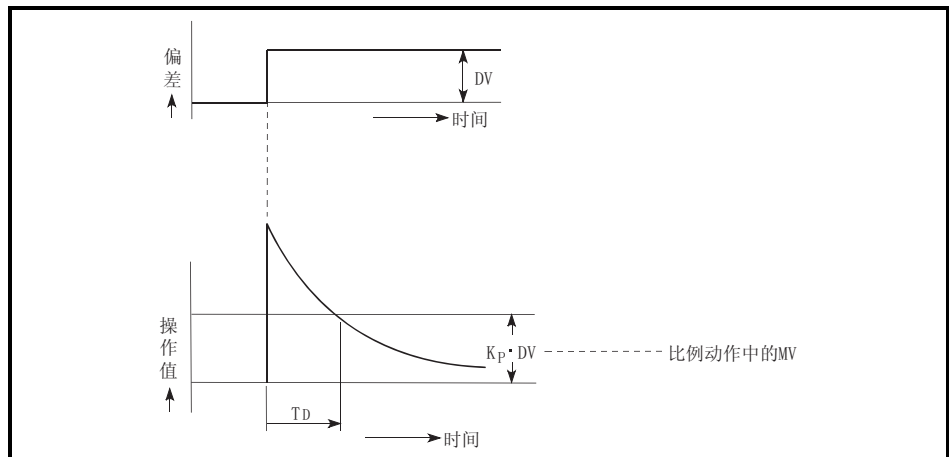


图 4.4 偏差为恒定值时的微分动作

- (4) 微分动作常与比例动作组合使用 (PD 动作) 或与比例及积分动作组合使用 (PID 动作)。  
微分动作不能独立使用。

备注

关于完全微分与不完全微分之间的区别。

[不完全微分]

不完全微分是指，在微分项的输入中加入了一个一次延迟过滤器的 PID 控制。

S.PIDCONT 指令是不完全微分 PID 控制指令。

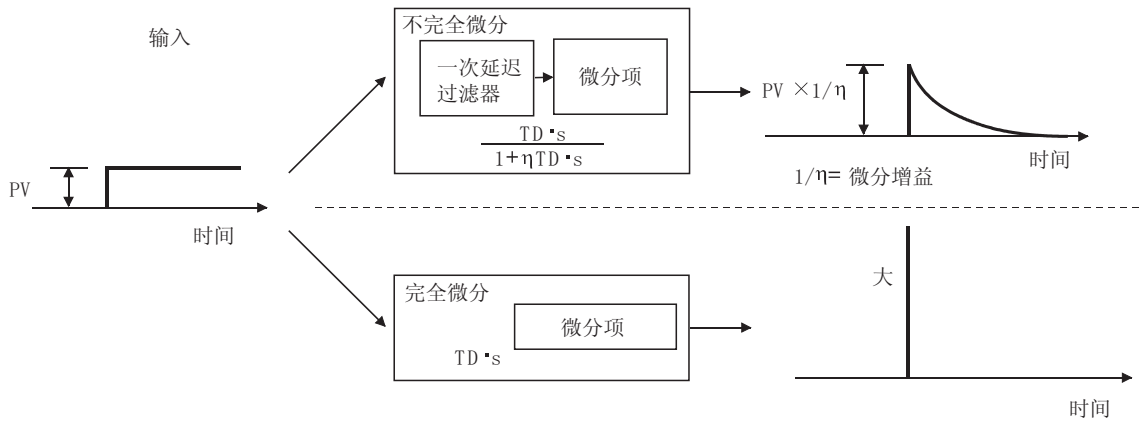
不完全微分在如下的情况下有效。

- 易受高频噪声影响的控制。
- 在完全微分方式中发生阶跃变化时不能提供有效能量使操作端动作时。

[完全微分]

完全微分是指原样不变地使用微分项的输入的 PID 控制。

PIDCONT 指令是完全微分 PID 控制指令。



## 4. PID 控制的功能

### 4.2.6 PID 动作

以下介绍将比例动作(P 动作)、积分动作(I 动作)和微分动作(D 动作)组合使用时的控制方法。

(1) PID 动作期间, 系统通过由(P+I+D)动作计算出来的 MV(操作值)进行控制。

(2) 恒定偏差的阶跃响应时的 PID 动作如图 4.5 所示。

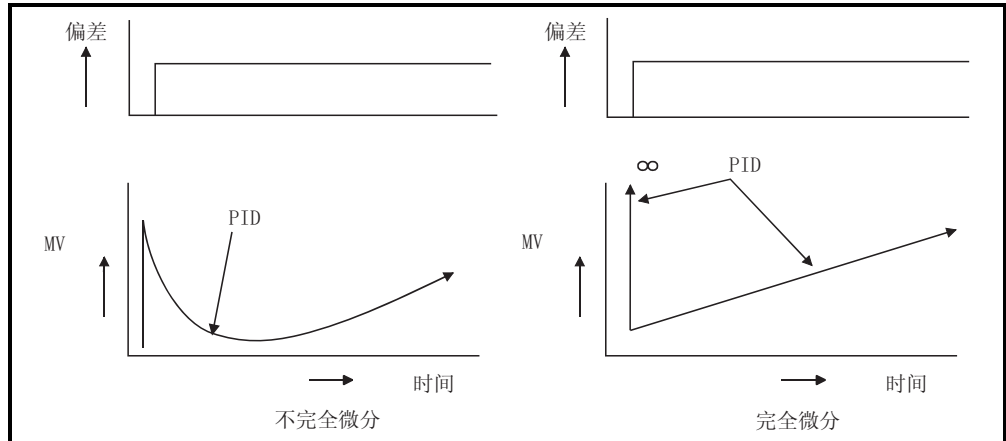


图 4.5 恒定偏差的 PID 动作

### 4.3 其它功能

以下介绍在使用 PID 控制指令的 PID 控制中, 通过无冲击切换功能自动执行 MV 上限/下限控制等有关内容。

## 4. PID 控制的功能

### 4.3.1 无冲击切换功能

- (1) 无冲击切换是指，当控制模式在手动与自动之间相互切换时，可连续控制 MV (操作值) 的功能。
- (2) 当切换模式 (在手动与自动之间) 时, 数据在“自动模式中的 MV 区 (自动 MV)”与“手动模式中的 MV 区 (手动 MV)”之间传送, 有关内容如下所示。  
在 I/O 数据区切换控制模式。(参阅 5.2 节)。
  - (a) 从手动模式切换到自动模式..... 将手动模式下的 MV 传送到自动模式下的 MV 区中。
  - (b) 从自动模式切换到手动模式..... 将自动模式下的 MV 传送到手动模式下的 MV 区中。

要点
(1) PID 控制的手动模式和自动模式。 <ol style="list-style-type: none"><li>1) 自动模式 用 PID 控制指令执行 PID 运算。 根据计算出的 MV 对控制对象进行控制。</li><li>2) 手动模式 不执行 PID 运算。根据用户计算出的 MV 对控制对象进行控制。</li></ol>
(2) 对于设置成手动模式的环路，将各个采样周期的 PV (测定值) 存储到设置值区中。

### 4.3.2 MV 上限/下限控制功能

- (1) MV 上限/下限控制功能是指，对在 PID 运算中计算出的 MV 的上限或下限进行控制的功能。该功能只在自动模式中有效，不能在手动模式中执行该操作。
- (2) 通过设置 MV 的上限 (MVHL) 和下限 (MVLL)，可以将 PID 运算中计算出的 MV 控制在该上下限范围以内。

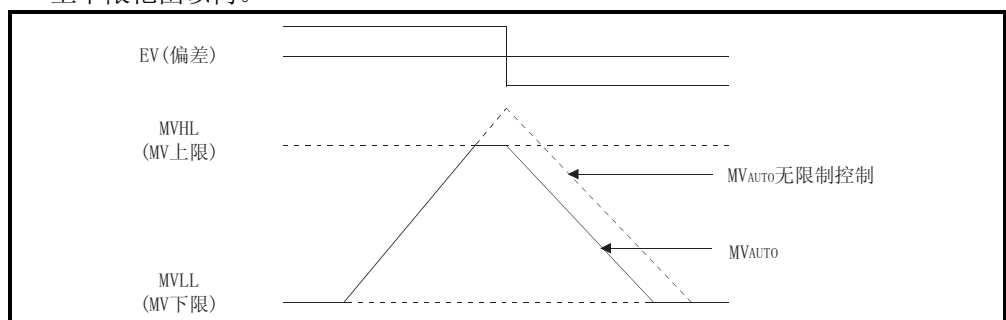


图 4.6 根据 MV 上限/下限的动作

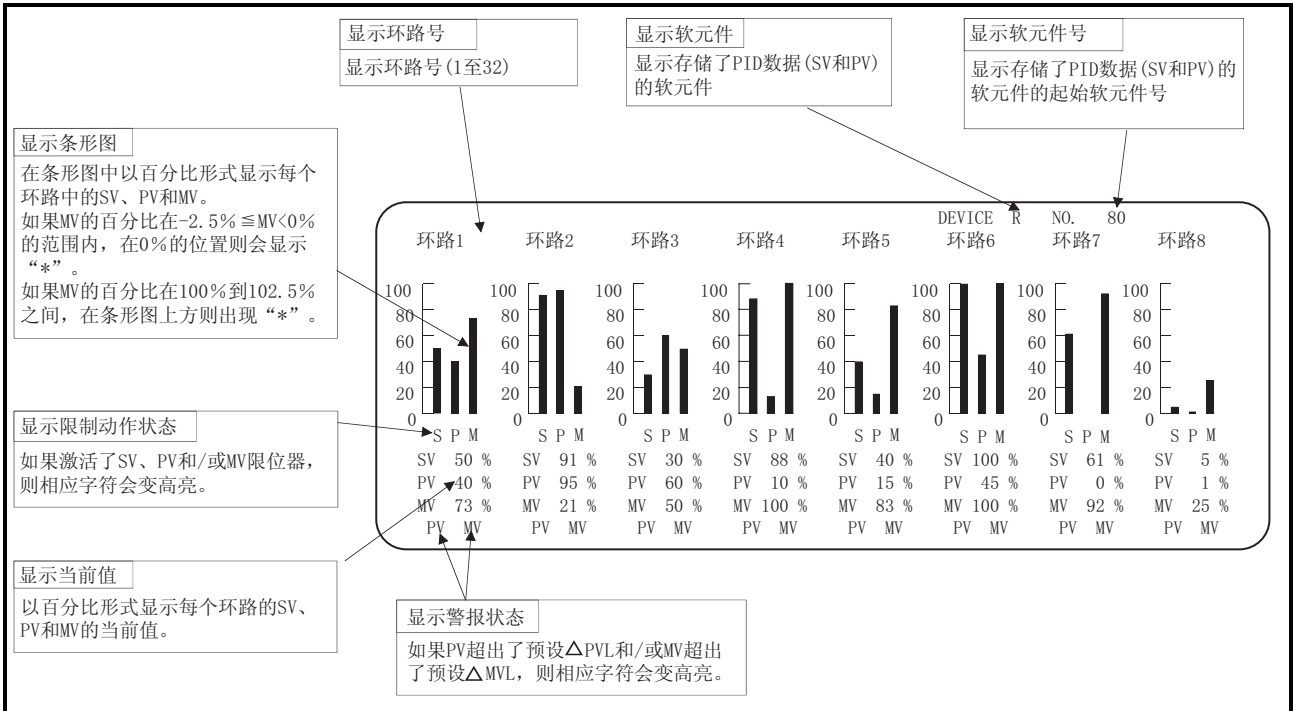
- (3) 当使用 MV 上限/下限控制时，按上图所示进行 MV 控制。  
MVHL (MV 上限) 和 MVLL (MV 下限) 的值应设置在 -50 至 2050 之间或由用户任意设置 (QnACPU 除外)。  
默认值的设置如下所示：
  - 上限 ..... 2000 (或用户设置值)
  - 下限 ..... 0 (或用户设置值)设置为上限的值不能小于设置为下限的值。  
否则将会出现错误。

## 4. PID 控制的功能

### 4.3.3 通过 AD57 (S1) 监视 PID 控制(仅用于 QnACPU)

可以用 AD57 (S1) 型 CRT 控制模块以条形图方式监视 PID 控制的运算结果。

(1) 监视画面显示从指定环路号开始的八个环路的监视信息。



#### 要点

SV、PV 和 MV 的当前显示是以 2000 为基准的%显示。

- 1) SV 的百分比显示.....  $\frac{SV}{2000} \times 100(\%)$
- 2) PV 的百分比显示.....  $\frac{PV}{2000} \times 100(\%)$
- 3) MV 的百分比显示.....  $\frac{MV}{2000} \times 100(\%)$

(2) 通过 PID57 指令执行 AD57 (S1) 的监视。

关于 PID57 指令的详细内容请参阅 9.1.3 项。

## 4. PID 控制的功能

### 4.3.4 将手动模式下的 PV 传送到 SV 存储软元件的功能

当使用 PID 控制指令执行 PID 控制时，在手动模式时也应执行 PID 运算指令。

在手动模式下执行 PID 运算指令时，可以通过 PID 无冲击处理标志 (SM774、SM794) 的 ON/OFF 状态，选择是否将 A/D 转换模块中获取的 PV 传送到 SV 存储软元件中。

PID 无冲击处理标志		动作内容
SM794 (不完全微分)	SM774 (完全微分)	
OFF		<ul style="list-style-type: none"> <li>• 执行 PID 运算指令时，将 PV 传送到 SV 存储软元件中。</li> <li>• 从手动模式切换到自动模式时，继续执行手动模式时的 MV 输出。</li> <li>• 切换到自动模式后如果改变 SV，进行从手动模式时输出的 MV 变为 SV 的控制。</li> </ul>
ON		<ul style="list-style-type: none"> <li>• 执行 PID 运算指令时，不将 PV 传送到 SV 存储软元件中。</li> <li>• 从手动模式切换为自动模式时，进行从手动模式的 MV 输出变为 SV 的控制。</li> <li>• 在切换为自动模式之前，事先将 SV 存储到 SV 存储软元件中。</li> </ul>

#### 要点

从手动模式切换为自动模式时，根据 SM774/SM794 的 ON/OFF 状态，其控制有下列差别：

- 当 SM774/SM794 为 OFF 时，由于将 PV 传送到 SV 存储软元件中，所以从手动模式切换为自动模式时，PV 与 SV 之间无偏差。  
因此，在切换时不会发生 MV 的骤变。  
相反，因为切换模式后的 SV 与自动模式下的目标值不同，所以须由用户通过顺控程序逐步将 SV 变更为目标值。
- 当 SM774/SM794 为 ON 时，由于不将 PV 传送到 SV 存储软元件中。因此，从手动模式切换到自动模式时，PV 与 SV 之间存在偏差。  
如果在切换模式时偏差较大，有可能会发生 MV 的骤变。  
应将该方法用于进行 PV 与 SV 相近时的模式切换系统中。  
该方法可以无需通过顺控程序逐步地将 SV 变更成目标值而立即在自动模式下执行 PID 控制。

#### 备注

- 通过 PID 运算指令可以将 SV 和 PV 存储到指定为 I/O 数据区的软元件中。

## 4. PID 控制的功能

### 4.3.5 PID 控制数据及 I/O 设置范围的变更功能 (用于 QCPU、LCPU)

可以由用户任意更改 PID 控制数据 (参阅 5.1 节) 和 I/O 数据 (参阅 5.2 节) 中下列数据的设置范围。

项目	设置数据
PID 控制数据	MV 下限值
	MV 上限值
	MV 变化率限制值
	PV 变化率限制值
I/O 数据	SV
	PV
	自动 MV
	过滤后的 PV
	手动 MV

为了使用户设置生效，应将用于 PID 上下限制设置的特殊寄存器 (SD774、SD775、SD794、SD795) 中相应环路的对应位设置为“1”。

PID 上下限制设置用特殊寄存器		设置范围
不完全微分	完全微分	
SD794	SD774	<p>                     b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0                      0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1                      ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑                      环路15 环路16 环路8 环路1 环路2                 </p>
SD795	SD775	<p>                     b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0                      0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/1                      ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑                      环路31 环路32 环路17 环路18                 </p>

0: 有 PID 上下限制 (系统固定值)

1: 无 PID 上下限制 (用户设置)

#### 要点

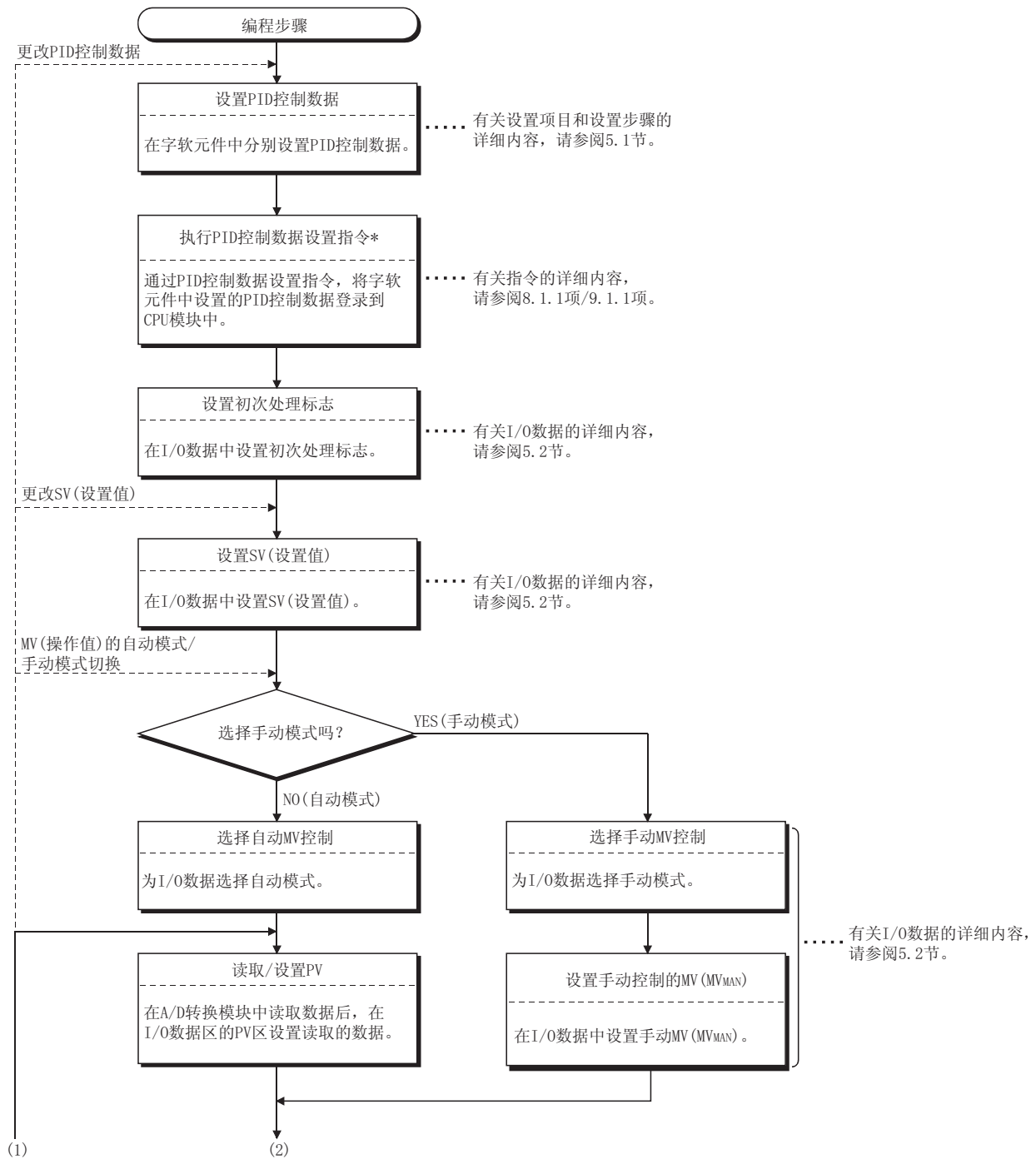
基本型 QCPU 有 8 个环路。  
SD774 和 SD794 的 b0 至 b7 有效。





## 5. PID 控制步骤

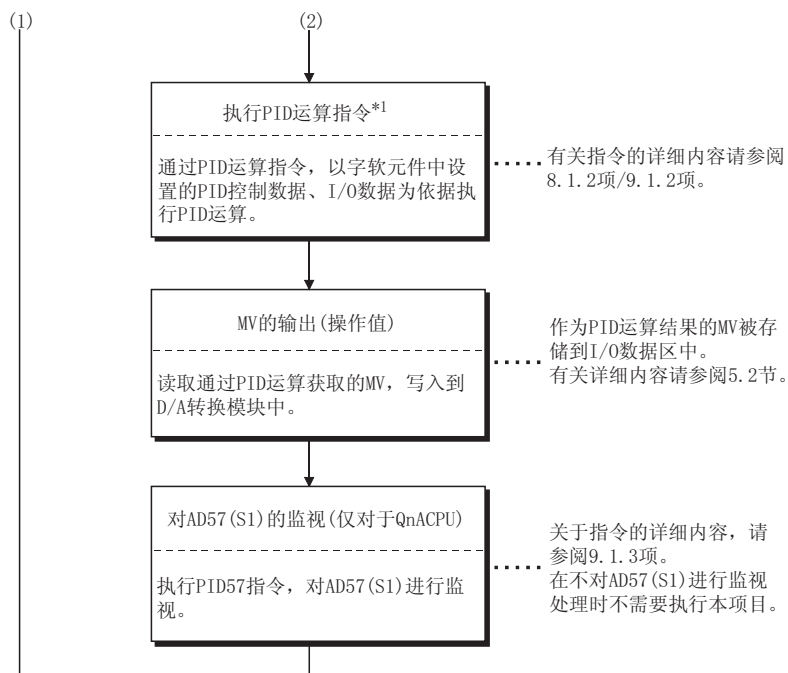
执行 PID 控制必需的编程步骤如下所示。



### 备注

\*: 下列指令可用作 PID 控制数据设置指令。

- S. PIDINIT(不完全微分)
- PIDINIT(完全微分)



**要点**

- 在顺控制程序的各个扫描中即使登录或更改PID控制数据也不会出现问题。但是，当登录或更改了PID控制数据时应执行PID控制用数据设置指令\*2。如果未执行PID控制数据设置指令的指令，在执行PID运算指令时将不能反应出已登录或修改的PID控制用数据。
- 在每个环路中使用参数变更指令\*3变更PID控制数据时，不必执行PID控制用数据设置指令。

**备注**

- \*1: 下列指令可用为PID运算指令。
  - S. PIDCONT(不完全微分)
  - PIDCONT(完全微分)
- \*2: 下列指令可用为PID控制数据设置指令。
  - S. PIDINIT(不完全微分)
  - PIDINIT(完全微分)
- \*3: 下列指令可用为参数变更指令。
  - S. PIDPRMW(不完全微分)
  - PIDPRMW(完全微分)

# 备忘录

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

## 5. PID 控制步骤

### 5.1 PID 控制用数据

(1) PID 控制用数据为 PID 运算中用于设置参照值设置的数据。

执行 PID 运算指令\*1 之前，用 PID 控制数据设置指令\*2 将 PID 控制用数据登录到 CPU 模块中。

PID 控制数据分为两种类型：“所有环路通用的设置数据”和“单个环路的设置数据”。

(a) 对于基本型 QCPU

表 5.1 PID 控制用数据列表

	数据号	数据项目	内容	不完全微分			
				有 PID 上下限制		无 PID 上下限制	
				设置范围	用户指定范围	设置范围	用户指定范围
通用设置数据	1	使用环路数量	设置要执行 PID 运算的环路数量。	1 至 8	1 至 8	1 至 8	1 至 8
	2	一次扫描中的执行环路数	设置当多个环路达到采样周期时间时，一次 PID 运算中执行的环路数。	1 至 8	1 至 8	1 至 8	1 至 8
各个环路的数据	1	选择运算公式	选择 3.1.2 节/3.2.2 节中的 PID 运算公式。	正向动作: 0 逆向动作: 1	0 或 1	正向动作: 0 逆向动作: 1	0 或 1
	2	采样周期(Ts)	设置 PID 运算的执行周期	0.01 至 60.00 s	1 至 6000 (单位: 10ms)	0.01 至 60.00 s	1 至 6000 (单位: 10ms)
	3	比例常数(Kp)	PID 运算的比例增益	0.01 至 100.00	1 至 10000 (单位: 0.01)	0.01 至 100.00	1 至 10000 (单位: 0.01)
	4	积分常数(Ti)	该常数表示积分动作(I 动作)效果的大小。增大积分常数时将减慢操作值的变化。	0.1 至 3000.0s  无限大(∞) 〔如果Ti设定超过3000.0 s〕	1 至 32767 (单位: 100ms)	0.1 至 3000.0 s  无限大(∞) 〔如果Ti设定超过3000.0s〕	1 至 32767 (单位: 100ms)
	5	微分常数(Td)	该常数表示微分动作(D 动作)效果的大小。增大微分常数时控制对象的轻微变化也会导致操作值的显著变化。	0.00 至 300.00s	0 至 30000 (单位: 10ms)	0.00 至 300.00s	0 至 30000 (单位: 10ms)
	6	过滤器系数(α)	设置对于测定值的过滤度。系数值愈趋近于 0 过滤效果愈小直至消失。	0 至 100%	0 至 100	0 至 100%	0 至 100

## 5. PID 控制步骤

### 备注

\*1: 下列指令可作为 PID 运算指令。

- S.PIDCONT(不完全微分)
- PIDCONT(完全微分)

\*2: 下列指令可作为 PID 控制用数据设置指令。

- S.PIDINIT(不完全微分)
- PIDINIT(完全微分)

完全微分					设置数据超出指定范围时的处理
有 PID 上下限制		无 PID 上下限制			
设置范围	用户指定范围	设置范围	用户指定范围		
1 至 8	1 至 8	1 至 8	1 至 8	出现错误, 不执行所有环路的 PID 运算。	
1 至 8	1 至 8	1 至 8	1 至 8		
正向动作: 0 逆向动作: 1	0 或 1	正向动作: 0 逆向动作: 1	0 或 1	出现错误, 不执行相应环路的 PID 运算。	
0.01 至 60.00s	1 至 6000 (单位: 10ms)	0.01 至 60.00 s	1 至 6000 (单位: 10ms)		
0.01 至 100.00	1 至 10000 (单位: 0.01)	0.01 至 100.00	1 至 10000 (单位: 0.01)		
0.1 至 3000.0s	1 至 32767 (单位: 100ms)	0.1 至 3000.0s	1 至 32767 (单位: 100ms)	出现错误, 不执行相应环路的 PID 运算。	
无限大( $\infty$ ) 〔如果 $T_i$ 设定超过3000.0 s〕		无限大( $\infty$ ) 〔如果 $T_i$ 设定超过3000.0 s〕			
0.00 至 300.00 s	0 至 30000 (单位: 10ms)	0.00 至 300.00s	0 至 30000 (单位: 10ms)	出现错误, 不执行相应环路的 PID 运算。	
0 至 100%	0 至 100	0 至 100%	0 至 100		

表 5.1 PID 控制数据列表

	数据号	数据项目	内容	不完全微分			
				有 PID 上下限制		无 PID 上下限制	
				设置范围	用户指定范围	设置范围	用户指定范围
各个环路的数据	7	MV 下限 (MVLL)	在自动模式下, 为在 PID 运算中计算出的 MV (操作值) 设置下限。当 MV 低于 MV 下限值时, 将 MVLL 用作 MV。	-50 至 2050	-50 至 2050	-32768 至 32767	-32768 至 32767
	8	MV 上限 (MVHL)	在自动模式下, 为在 PID 运算中计算出的 MV 设置上限。当 MV 超出 MV 上限值时, 将 MVHL 用作 MV。	-50 至 2050	-50 至 2050	-32768 至 32767	-32768 至 32767
	9	MV 变化率限制 ( $\Delta$ MVL)	为上一个 MV 和当前 MV 之间的变化量设置限制值。当 MV 变化量超出限制值时, 报警用软元件的位 1 (b1) 将变为 1。不限制 MV 变化量 (即使 MV 变化量超出了限制值, 也将其作为计算 MV 的 MV 变化量保持不变)。	0 至 2000	0 至 2000	-32768 至 32767	-32768 至 32767
	10	PV 变化率限制 ( $\Delta$ PVL)	为上一个 PV 和当前 PV 之间的变化量设置限制值。当 PV 变化量超出限制值时, 报警用软元件的位 0 (b0) 将变为 1。不限制 PV 变化量。(即使 PV 变化量超出了限制值, 也将其作为执行 PID 运算的 PV 变化量保持不变。)	0 至 2000	0 至 2000	-32768 至 32767	-32768 至 32767
	11	微分增益 ( $K_D$ )	为微分动作设置时间宽度 (动作延迟)。值愈大时间宽度愈小, 动作更接近于完全微分。	0.00 至 300.00 (理想值为 8.00) 无限大 ( $\infty$ ) (如果 $K_D$ 设定超过 300.00)	0 至 32767 (单位: 0.01)	0.00 至 300.00 (理想值为 8.00) 无限大 ( $\infty$ ) (如果 $K_D$ 设定超过 300.00)	0 至 32767 (单位: 0.01)

## 5. PID 控制步骤

完全微分					设置数据超出指定范围时的处理
有 PID 上下限制		无 PID 上下限制			
设置范围	用户指定范围	设置范围	用户指定范围		
-50 至 2050	-50 至 2050	-32768 至 32767	-32768 至 32767		在“有 PID 上下限制”的情况下，进行了下列值的转换后，执行 PID 运算。 • 当 MVLL 或 MVHL 值低于 -50 时，转换为“-50”。 • 当 MVLL 或 MVHL 值超出 2050 时，转换为“2050”。
-50 至 2050	-50 至 2050	-32768 至 32767	-32768 至 32767		
0 至 2000	0 至 2000	-32768 至 32767	-32768 至 32767		在“有 PID 上下限制”的情况下，进行了下列值的转换后，执行 PID 运算。 • 当 $\Delta$ MVL 值低于 0 时，将其转换为 0。 • 当 $\Delta$ MVL 值超出 2000 时，将其转换为 2000。
0 至 2000	0 至 2000	-32768 至 32767	-32768 至 32767		在“有 PID 上下限制”的情况下，进行了下列值的转换后，执行 PID 运算。 • 当 $\Delta$ PVL 值低于 0 时，将其转换为 0。 • 当 $\Delta$ PVL 值超出 2000 时，将其转换为 2000。
-	-	-	-		出现错误，不执行相应环路的 PID 运算。

## 5. PID 控制步骤

(b) 对于高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU

表 5.2 PID 控制用数据列表

	数据号	数据项目	内容	不完全微分			
				有 PID 上下限制		无 PID 上下限制	
				设置范围	用户指定范围	设置范围	用户指定范围
通用设置数据	1	环路数量	设置要执行 PID 运算的环路数量。	1 至 32	1 至 32	1 至 32	1 至 32
	2	一次扫描中的执行环路数量	设置当多个环路达到采样周期时间时，一次 PID 运算中执行的环路数量。	1 至 32	1 至 32	1 至 32	1 至 32
各个环路的数据	1	选择运算公式	选择 3.1.2 项/3.2.2 项中的 PID 运算公式。	正向动作: 0 逆向动作: 1	0 或 1	正向动作: 0 逆向动作: 1	0 或 1
	2	采样周期(Ts)	设置 PID 运算的执行周期。	0.01 至 60.00s	1 至 6000 (单位: 10ms)	0.01 至 60.00s	1 至 6000 (单位: 10ms)
	3	比例常数(Kp)	PID 运算的比例增益。	0.01 至 100.00	1 至 10000 (单位: 0.01)	0.01 至 100.00	1 至 10000 (单位: 0.01)
	4	积分常数(Ti)	该常数表示积分动作(I 动作)效果的大小。增加积分常数将减慢操作值的变化。	0.1 至 3000.0 s	1 至 32767 (单位: 100ms)	0.1 至 3000.0 s	1 至 32767 (单位: 100ms)
				无限大( $\infty$ ) (如果Ti设定超过3000.0s)		无限大( $\infty$ ) (如果Ti设定超过3000.0s)	
	5	微分常数(Td)	该常数表示微分动作(D 动作)效果的大小。增大微分常数时控制对象的轻微变化也会导致操作值的显著变化。	0.00 至 300.00s	0 至 30000 (单位: 10ms)	0.00 至 300.00s	0 至 30000 (单位: 10ms)
6	过滤器系数( $\alpha$ )	设置对于测定值的过滤度。系数值愈趋近于 0，过滤效果愈小直至消失。	0 至 100 %	0 至 100	0 至 100 %	0 至 100	



## 5. PID 控制步骤

完全微分					设置数据超出指定范围时的处理
有 PID 上下限限制		无 PID 上下限限制			
设置范围	用户指定范围	设置范围	用户指定范围		
1 至 32	1 至 32	1 至 32	1 至 32		出现错误，不执行所有环路的 PID 运算。
1 至 32	1 至 32	1 至 32	1 至 32		
正向动作: 0 逆向动作: 1	0 或 1	正向动作: 0 逆向动作: 1	0 或 1		出现错误，不执行相应环路的 PID 运算。
0.01 至 60.00 s	1 至 6000 (单位: 10ms)	0.01 至 60.00 s	1 至 6000 (单位: 10ms)		
0.01 至 100.00	1 至 10000 (单位: 0.01)	0.01 至 100.00	1 至 10000 (单位: 0.01)		
0.1 至 3000.0 s	1 至 32767 (单位: 100ms)	0.1 至 3000.0 s	1 至 32767 (单位: 100ms)		出现错误，不执行相应环路的 PID 运算。
无限大( $\infty$ ) 〔如果T <sub>i</sub> 设定超过3000.0 s〕		无限大( $\infty$ ) 〔如果T <sub>i</sub> 设定超过3000.0 s〕			
0.00 至 300.00 s	0 至 30000 (单位: 10ms)	0.00 至 300.00 s	0 至 30000 (单位: 10ms)		出错错误，不执行相应环路的 PID 运算。
0 至 100 %	0 至 100	0 至 100 %	0 至 100		

表 5.2 PID 控制数据列表

	数据号	数据项目	说明	不完全微分			
				有 PID 上下限限制		无 PID 上下限限制	
				设置范围	用户指定范围	设置范围	用户指定范围
各个环路的数据	7	MV 下限 (MVLL)	在自动模式下, 为在 PID 运算中计算出的 MV (操作值) 设置下限。当 MV 低于 MV 下限时, 将 MVLL 用作 MV。	-50 至 2050	-50 至 2050	-32768 至 32767	-32768 至 32767
	8	MV 上限 (MVHL)	在自动模式下, 为在 PID 运算中计算出的 MV 设置上限。当 MV 超出 MV 上限时, 将 MVHL 用作 MV。	-50 至 2050	-50 至 2050	-32768 至 32767	-32768 至 32767
	9	MV 变化率限制 ( $\Delta$ MVL)	为上一个 MV 和当前 MV 之间的变化量设置限制值。当 MV 变化量超出限制值时, 报警用软元件位 1 (b1) 将变为 1。不限制 MV 变化量。(即使 MV 变化量超出了限制值, 也将其作为计算 MV 的 MV 变化量保持不变。)	0 至 2000	0 至 2000	-32768 至 32767	-32768 至 32767
	10	PV 变化率限制 ( $\Delta$ PVL)	为上一个 PV 和当前 PV 之间的变化量设置限制。当 PV 变化量超出限制值时, 报警用软元件的位 0 (b0) 将变为 1。不限制 PV 变化量。(即使 PV 变化量超出了限制值, 也将其作为执行 PID 运算的 PV 变化量保持不变。)	0 至 2000	0 至 2000	-32768 至 32767	-32768 至 32767
	11	微分增益 (K <sub>D</sub> )	为微分动作设置时间周期 (动作延迟)。值愈大时, 时间宽度愈小, 动作更接近于完全微分。	0.00 至 300.00 (理想值为 8.00) 无限大 ( $\infty$ ) 如果 K <sub>D</sub> 设定超过 300.00	0 至 32767 (单位: 0.01)	0.00 至 300.00 (理想值为 8.00) 无限大 ( $\infty$ ) 如果 K <sub>D</sub> 设定超过 300.00	0 至 32767 (单位: 0.01)

## 5. PID 控制步骤

完全微分					设置数据超出指定范围时的处理
有 PID 上下限制		无 PID 上下限制			
设置范围	用户指定范围	设置范围	用户指定范围		
-50 至 2050	-50 至 2050	-32768 至 32767	-32768 至 32767	在高性能型 QCPU 的“有 PID 上下限制”时或在 QnACPU 中，进行了下列值的转换后，执行 PID 运算。 <ul style="list-style-type: none"> <li>当 MVLL 或 MVHL 值低于-50 时，将其转换为-50。</li> <li>当 MVLL 或 MVHL 值超出 2050 时，将其转换为 2050。</li> </ul>	
-50 至 2050	-50 至 2050	-32768 至 32767	-32768 至 32767		
0 至 2000	0 至 2000	-32768 至 32767	-32768 至 32767	在“有 PID 上下限制”的情况下，进行了下列值转换后，执行 PID 运算。 <ul style="list-style-type: none"> <li>当 <math>\Delta</math>MVL 低于 0 时，将其值转换为 0。</li> <li>当 <math>\Delta</math>MVL 值高于 2000 时，将其转换为 2000。</li> </ul>	
0 至 2000	0 至 2000	-32768 至 32767	-32768 至 32767	在“有 PID 上下限制”的情况下，进行了下列转换后，执行 PID 运算。 <ul style="list-style-type: none"> <li>当 <math>\Delta</math>PVL 值低于 0 时，将值转换为 0。</li> <li>当 <math>\Delta</math>PVL 值超出了 2000 时，将其转换为 2000。</li> </ul>	
-	-	-	-	出现错误，且不执行相应环路的 PID 运算。	

(c) 对于 QnACPU

表 5.3 PID 控制数据列表

数据号	数据项目	说明	设置范围	用户指定范围	设置数据超出指定范围时的处理	
设置通用数据	1	环路数量	设置要执行 PID 运算的环路数量。	1 至 32	1 至 32	出现错误，不执行所有环路的 PID 运算。
	2	一次扫描中执行的环路数量	设置多个环路达到采样周期时间时，一次 PID 运算中执行的环路数量。	1 至 32	1 至 32	
各个环路的数据	1	选择运算公式	选择 3.2.2 项中的 PID 运算公式。	正向动作: 0 逆向动作: 1	0 或 1	出现错误，不执行相应环路的 PID 运算。
	2	采样周期 (Ts)	设置 PID 运算的执行周期。	0.01 至 60.00s	1 至 6000 (单位:10ms)	
	3	比例常数 (Kp)	PID 运算的比例增益	0.01 至 100.00	1 至 10000 (单位:0.01)	
	4	积分常数 (Ti)	该常数表示积分动作 (I 动作) 效果的大小。 增大积分常数时将减慢操作值的变化。	0.1 至 3000.0s 无限大 (∞) 如果 Ti 设定超过 3000.0s	1 至 32767 (单位:100ms)	出现错误，不执行相应环路的 PID 运算。
	5	微分常数 (Td)	该常量表示微分动作 (D 动作) 效果的大小。 增大微分常数时控制对象的轻微变化也会导致操作值的显著变化。	0.00 至 300.00s	0 至 30000 (单位:10ms)	出现错误，不执行相应环路的 PID 运算。
	6	过滤器系数 (α)	设置对于测定值的过滤度。 系数值愈趋近于 0 过滤效果愈小直至消失。	0 至 100%	0 至 100	
	7	MV 下限 (MVLL)	在自动模式下，为在 PID 运算中计算出的 MV 设置下限。 当 MV 低于 MV 下限时，将 MVLL 用作 MV。	-50 至 2050	-50 至 2050	进行了下列值的转换后，执行 PID 运算。 • 当 MVLL 或 MVHL 值低于 -50 时，将其转换为 -50。 • 当 MVLL 或 MVHL 值超出 2050 时，将其转换为 2050。
	8	MV 上限 (MVHL)	在自动模式下，为在 PID 运算中计算出的 MV 设置上限。 当 MV 超出 MV 上限时，将 MVHL 用作 MV。	-50 至 2050	-50 至 2050	
	9	MV 变化率限制 (ΔMVL)	为上一个 MV 和当前 MV 之间的变化量设置限制值。 当 MV 变化量大于限制值时，报警用软元件的位 1 (b1) 将变为 1。 不限制 MV 变化量。 (即使 MV 变化量超出了限制值，也将其作为计算 MV 的 MV 变化量保持不变。)	0 至 2000	0 至 2000	进行了下列值的转换后，执行 PID 运算。 • 当 ΔMVL 值低于 0 时，将其转换为 0。 • 当 ΔMVL 值超出 2000 时，将其值转换为 2000。
	10	PV 变化率限制 (ΔPVL)	为上一个 PV 和当前 PV 之间的变化量设置限制值。 当 PV 变化量大于限制值时，报警用软元件的位 0 (b0) 将变为 1。 不限制 PV 变化量。 (即使 PV 变化量超出了限制值，也将其作为执行 PID 运算的 PV 变化量保持不变。)	0 至 2000	0 至 2000	进行了下列值的转换后，执行 PID 运算。 • 当 ΔPVL 值低于 0 时，将其转换为 0。 • 当 ΔPVL 值超出 2000 时，将其转换为 2000。

## 5. PID 控制步骤

(2) 可以对 PID 控制用数据设置任意的字软元件号。

但是，必须将所有数据按所使用的环路数设置为连续的软元件号。

(3) PID 控制用数据的分配如下所示。

(a) 用于不完全微分时

指定的软元件号+0	环路数量	}	用于所有环路
+1	一次扫描中的环路数量		
+2	选择运算表达式	}	用于1号环路 (14个字)
+3	采样周期 (Ts)		
+4	比例常数 (Kp)		
+5	积分常数 (Ti)		
+6	微分常数 (Td)		
+7	过滤器系数 ( $\alpha$ )		
+8	MV 下限 (MVLL)		
+9	MV 上限 (MVHL)		
+10	MV 变化率限制 ( $\Delta$ MVL)		
+11	PV 变化率限制 ( $\Delta$ PVL)		
+12	0 *		
+13	微分增益 (Kd)		
+14	0 *		
+15	0 *		
+16	选择运算表达式	}	用于2号环路 (14个字)
+17	采样周期 (Ts)		
+18	比例常数 (Kp)		
+19	积分常数 (Ti)		
+20	微分常数 (Td)		
+21	过滤器系数 ( $\alpha$ )		
+22	MV 下限 (MVLL)		
+23	MV 上限 (MVHL)		
+24	MV 变化率限制 ( $\Delta$ MVL)		
+25	PV 变化率限制 ( $\Delta$ PVL)		
+26	0 *		
+27	微分增益 (Kd)		
+28	0 *		
+29	0 *		
至	至	}	使用的环路总数
+ (m+0)	选择运算表达式		
+ (m+1)	采样周期 (Ts)		
+ (m+2)	比例常数 (Kp)		
+ (m+3)	积分常数 (Ti)		
+ (m+4)	微分常数 (Td)		
+ (m+5)	过滤器系数 ( $\alpha$ )		
+ (m+6)	MV 下限 (MVLL)		
+ (m+7)	MV 上限 (MVHL)		
+ (m+8)	MV 变化率限制 ( $\Delta$ MVL)		
+ (m+9)	PV 变化率限制 ( $\Delta$ PVL)		
+ (m+10)	0 *		
+ (m+11)	微分增益 (Kd)		
+ (m+12)	0 *		
+ (m+13)	0 *		

$$m = (n-1) \times 14 + 2$$

### 要点

应将 0 存储于标有 “\*” 的 PID 控制数据区。

如果将 0 以外的数存储于标有 “\*” 的区域，将会出错且不执行处理。(出错代码：4100)

## 5. PID 控制步骤

(a) 应使用下面的公式计算设置 PID 控制用数据时要使用的软元件点数。

$$\text{软元件点数} = 2 + 14 \times n \quad (n: \text{使用的环路数量})$$

(b) 应以二进制数设置各个数据。

(c) 如果使用的环路数量的软元件点数超出指定软元件的最后软元件编号，将出错且不执行处理。(出错代码：4101)

(b) 用于完全微分时

指定的软元件号+0	环路数量	}	用于所有环路
+1	一次扫描中的环路数量		
+2	选择运算表达式	}	用于1号环路 (10个字)
+3	采样周期(Ts)		
+4	比例常数(Kp)		
+5	积分常数(Ti)		
+6	微分常数(Td)		
+7	过滤器系数( $\alpha$ )		
+8	MV下限(MVLL)		
+9	MV上限(MVHL)		
+10	MV变化率限制( $\Delta$ MVL)		
+11	PV变化率限制( $\Delta$ PVL)		
+12	选择运算表达式	}	用于2号环路 (10个字)
+13	采样周期(Ts)		
+14	比例常数(Kp)		
+15	积分常数(Ti)		
+16	微分常数(Td)		
+17	过滤器系数( $\alpha$ )		
+18	MV下限(MVLL)		
+19	MV上限(MVHL)		
+20	MV变化率限制( $\Delta$ MVL)		
+21	PV变化率限制( $\Delta$ PVL)		
+22	使用的环路总数	}	用于3号环路 (10个字)
+23	采样周期(Ts)		
至	至	}	用于n号环路 (10个字)
+(m+0)	选择运算表达式		
+(m+1)	采样周期(Ts)		
+(m+2)	比例常数(Kp)		
+(m+3)	积分常数(Ti)		
+(m+4)	微分常数(Td)		
+(m+5)	过滤器系数( $\alpha$ )		
+(m+6)	MV下限(MVLL)		
+(m+7)	MV上限(MVHL)		
+(m+8)	MV变化率限制( $\Delta$ MVL)		
+(m+9)	PV变化率限制( $\Delta$ PVL)		

m = (n-1) × 10 + 2

} 使用的环路总数

(a) 应使用下面的公式计算设置 PID 控制数据时要使用的软元件点数。

$$\text{软元件点数} = 2 + 10 \times n \quad (n: \text{使用的环路数量})$$

(b) 应以二进制数设置各个数据。

(c) 如果使用的环路数量的软元件点数超出指定软元件的最后软元件编号，将出错且不执行处理。(出错代码：4101)

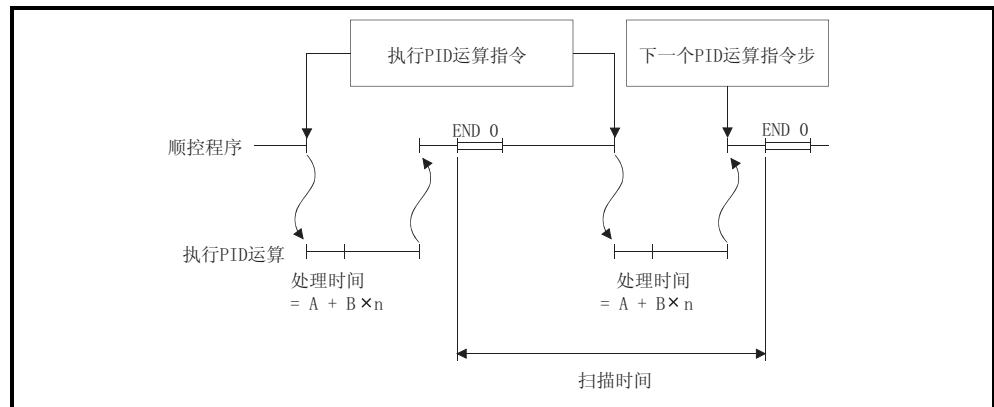
5.1.1 使用环路数和一次扫描的执行环路数

- (1) 使用环路数是指执行 PID 运算的环路数量。执行 PID 运算指令\*时，对所设置的环路数的采样周期进行计测。到达或超过采样周期的环路执行 PID 运算。
- (2) 执行 PID 运算指令时，处理时间与执行 PID 运算的环路数量成正比。

$$\text{处理时间} = A + B \times n$$

A: 用于采样周期计测的固定时间。  
 B: 一个环路执行 PID 运算需要的时间。  
 n= 环路数量

- (3) 一次扫描中执行的环路数是指：多个环路达到采样周期时，所设置的一次扫描中执行 PID 运算的环路数量。  
 如果预先设置了一次扫描中执行的环路数，则即使多个环路达到或超出采样周期，也只有所设置的环路数的环路执行 PID 运算指令。其余的环路在下次扫描中执行 PID 运算。



**要点**

如果达到或超出采样周期的环路数量大于一次扫描中执行的环路数量，执行 PID 运算的优先顺序如下所示：  
 (1) 编号越小环路其优先度越高。  
 (2) 如果存在上一次扫描中未执行 PID 运算的环路，这些环路具有最高的优先度。

**备注**

- \*: 下列指令可用作 PID 运算指令。
  - S. PIDCONT (不完全微分)
  - PIDCONT (完全微分)

## 5. PID 控制步骤

### 5.1.2 采样周期

(1) 采样周期就是执行 PID 运算的周期。

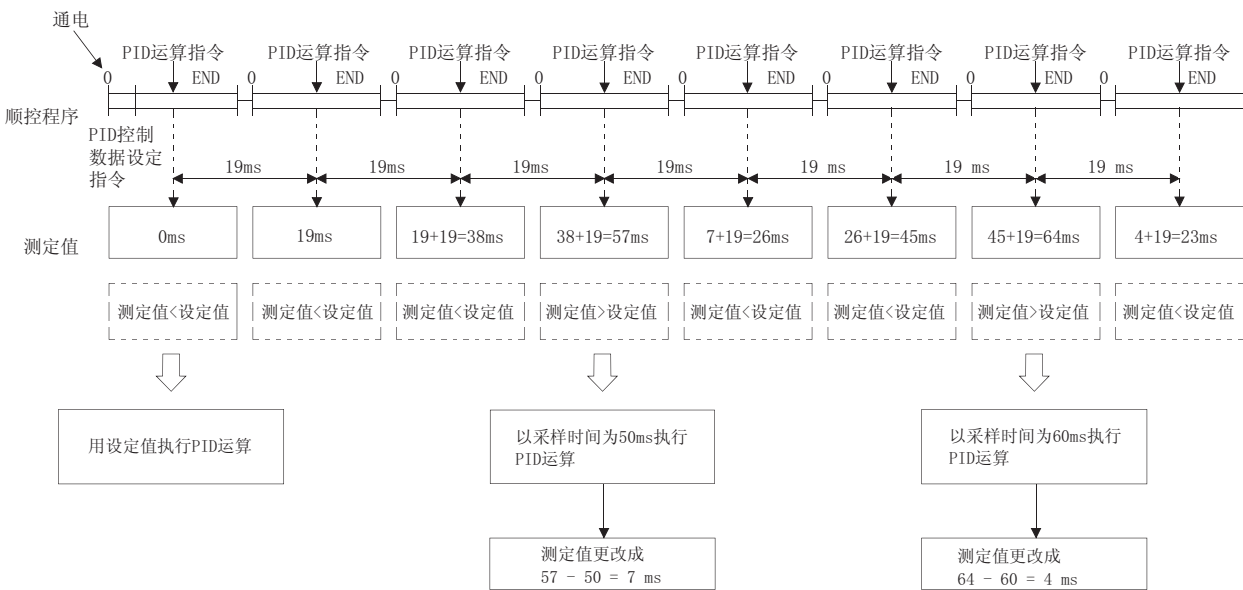
每次执行 PID 运算指令\*1 时，将一次扫描的测定时间累加到至上一次扫描为止的测定时间中。

当累计值达到或超出设置采样周期时，执行相应环路的 PID 运算。

(2) 用于 PID 运算的采样周期的测定值以 10ms 为单位进行舍取。

例如，如果采样周期设置为 50ms，测定值为 57ms，则以 50ms 为采样周期执行 PID 运算；如果测定值为 64ms，则以 60ms 为采样周期执行 PID 运算。

#### 当采样周期=50ms 时



#### 要点

执行 PID 运算指令时测定采样周期。因此，低于顺控程序扫描时间的值不能设置为采样周期。如果设置低于扫描时间的值为采样周期，将以扫描时间执行 PID 运算。

#### 备注

\*1: 下列指令可用为 PID 运算指令。

- S.PIDCONT (不完全微分)
- PIDCONT (完全微分)

\*2: 下列指令可用为 PID 控制数据设置指令。

- S.PIDINIT (不完全微分)
- PIDINIT (完全微分)





## 5. PID 控制步骤

### 5.2 I/O 数据


(1) I/O 数据包括用于执行 PID 运算的 SV(设置值)、PV(测定值)等输入数据和运算结果等的输出数据。

(2) I/O 数据区域分为“分配给各环路的数据区”和“用于执行 PID 运算的系统工作区”。

表 5.4 I/O 数据列表

数据名称		说明	设置范围	
			QCPU、LCPU	
			有 PID 上下限制	无 PID 上下限制
设置值	SV	• PID 控制的目标值	0 至 2000	-32768 至 32767
测定值	PV	• 从控制对象向 A/D 转换模块反馈的数据。	-50 至 2050	-32768 至 32767
自动操作值	MV	• 通过 PID 运算计算出的操作值。 • 从 D/A 转换模块输出到控制对象。	-50 至 2050	-32768 至 32767
过滤后的测定值	PVf	• 用 3.1.2 项要点(1)/3.2.2 项要点(1)中的运算公式计算出的测定值。	-50 至 2050	-32768 至 32767
手动操作值	MVMAN	• 在手动操作模式下, 存储从 D/A 转换模块输出的数据。	-50 至 2050	-32768 至 32767
手动/自动选择	MAN/AUTO	• 选择输出到 D/A 转换模块的数据是手动操作值还是自动操作值。 • 在手动控制模式下, 自动操作值保持不变。	0: 手动操作值 1: 自动操作值	0: 手动操作值 1: 自动操作值
报警	ALARM	• 用于确定 MV(操作值)和 PV(测定值)的变化率是否超出了允许范围。 • 一旦设置, 报警数据在用户复位之前保持不变。 • 如果 MV 超出了限制范围, 位 1(b1)将变为“1”。 • 如果 PV 超出了限制范围, 位 0(b0) 将变为“1”。		

## 5. PID 控制步骤

		当设置数据超出指定范围时的处理
	QnACPU	
	0 至 2000	对“有 PID 上下限限制”的 QCPU、LCPU 或 QnACPU，转换为下列值后执行 PID 运算。 <ul style="list-style-type: none"> <li>当 SV 低于 0 时，将 SV 转换为 0。</li> <li>当 SV 超出 2000 时，将 SV 转换为 2000。</li> </ul>
	-50 至 2050	对“有 PID 上下限限制”的 QCPU、LCPU 或 QnACPU，转换为下列值后执行 PID 运算。 <ul style="list-style-type: none"> <li>当 PV 低于 -50 时，将 PV 转换为 -50。</li> <li>当 PV 超出 2050 时，将 PV 转换为 2050。</li> </ul>
	-50 至 2050	-
	-50 至 2050	-
	-50 至 2050	对于“有 PID 上下限限制”的 QCPU、LCPU 或 QnACPU，转换为下列值后执行 PID 运算。 <ul style="list-style-type: none"> <li>当 MVMAN 低于 -50 时，将 MVMAN 转换为 -50。</li> <li>当 MVMAN 大于 2050 时，将 MVMAN 转换为 2050。</li> </ul>
	0: 手动操作值 1: 自动操作值	如果设置既不是 0 也不是 1 则会出现错误，不执行相应环路的 PID 运算。
	 <p>如果 PV 超出了限制范围，位 0 将变为“1”。</p> <p>如果 MV 超出了限制范围，位 1 将变为“1”。</p>	-

## 5. PID 控制步骤

(3) 可以将 I/O 数据指定为任意的字软元件号。但是，必须将相应环路的所有数据指定为连续的软元件号。

(4) I/O 数据分配如下所示。

(a) 对于不完全微分

指定的软元件号 +0	初次处理标志	写入		
+1	用于PID控制的工作区 (不可使用)	读取/写入 禁止		
至				
+9				
+10	设置值(SV)	写入	环路1的I/O 数据区 (23个字)	
+11	测定值(PV)	读取		
+12	自动操作值(MV)			
+13	过滤后的测定值(PVf)	写入		
+14	手动操作值(MV <sub>MAN</sub> )			
+15	手动/自动选择(MAN/AUTO)			
+16	报警(ALARM)	读取/写入		
+17	环路1的工作区 (不可使用)	读取/写入 禁止		
至				
+32				
+33	设置值(SV)	写入	环路2的I/O 数据区 (23个字)	
+34	测定值(PV)	读取		
+35	自动操作值(MV)			
+36	过滤后的测定值(PVf)	写入		
+37	手动操作值(MV <sub>MAN</sub> )			
+38	手动/自动选择(MAN/AUTO)			
+39	报警(ALARM)	读取/写入		
+40	环路2的工作区 (不可使用)	读取/写入 禁止		
至				
+55				
+56	设置值(SV)	写入	环路3的I/O 数据区 (23个字)	
+57	测定值(PV)	读取		
+58	自动操作值(MV)			
+59	至			
至				
至				
+ (m+0)	设置值(SV)	写入	环路n的I/O 数据区 (23个字)	
+ (m+1)	测定值(PV)	读取		
+ (m+2)	自动操作值(MV)			
+ (m+3)	过滤后的测定值(PVf)	写入		
+ (m+4)	手动操作值(MV <sub>MAN</sub> )			
+ (m+5)	手动/自动选择(MAN/AUTO)			
+ (m+6)	报警(ALARM)	读取/写入		
+ (m+7)	环路n的工作区 (不可使用)	读取/写入 禁止		
至				
+ (m+22)				

$m = (n-1) \times 23 + 10$

1) 应使用下面的公式计算输入/输出数据设置时使用的软元件点数:

$$\text{软元件点数} = 10 + 23 \times n \quad (n: \text{使用的环路数量})$$

2) 应以二进制设置各个数据。

- 3) 初次处理标志是对 PID 运算开始时处理方法进行设置。
  - 在初次 PID 运算处理时，被视为已到达所设置的采样周期后进行运算。
  - 对初次处理标志进行以下设置。
    - 0 ..... 使用的环路数量的 PID 运算在一次扫描中被成批处理。
    - 0 以外 ..... 使用的环路数量的 PID 运算被分割在多次扫描中处理。
      - 从完成初始次处理的环路开始依次进行采样。
      - 每次扫描中处理环路的数量是所设置的一次扫描执行的环路数量。
  
- 4) 对于 I/O 数据区中被指定为“写入”的数据应由用户通过顺控程序进行写入。

对于 I/O 数据区中被指定为“读取”的数据应由用户通过顺控程序读取后使用。

对于被指定为“读取/写入禁止”或“读取”的数据绝对不要进行写入。否则将不能正常进行 PID 运算。
  
- 5) 如果使用的环路数量的软元件点数超出指定软元件的最后软元件号，将会出错且不执行处理。(出错代码：4101)

(b) 对于完全微分

指定的软元件号	+0	初次处理标志	写入		
	+1	用于PID控制的工作区 (不可使用)	读取/写入 禁止		
	+9				
	+10	设置值(SV)	写入	环路1的I/O 数据区 (18个字)	
	+11	测定值(PV)			
	+12	自动操作值(MV)	读取		
	+13	过滤后的测定值(PVf)			
	+14	手动操作值(MV <sub>MAN</sub> )	写入		
	+15	手动/自动选择(MAN/AUTO)			
	+16	报警(ALARM)	读取/写入		
	+17 至 +27	环路1的工作区 (不可使用)	读取/写入 禁止		
	+28	设置值(SV)	写入		环路2的I/O 数据区 (18个字)
	+29	测定值(PV)			
	+30	自动操作值(MV)	读取		
	+31	过滤后的测定值(PVf)			
	+32	手动操作值(MV <sub>MAN</sub> )	写入		
	+33	手动/自动选择(MAN/AUTO)			
	+34	报警(ALARM)	读取/写入		
	+35 至 +45	环路2的工作区 (不可使用)	读取/写入 禁止		
	+46	设置值(SV)	写入	环路3的I/O 数据区 (18个字)	
	+47	测定值(PV)			
	+48	自动操作值(MV)	读取		
	+49 至	至			
	+(m+0)	设置值(SV)	写入	环路n的I/O 数据区 (18个字)	
	+(m+1)	测定值(PV)			
	+(m+2)	自动操作值(MV)	读取		
	+(m+3)	过滤后的测定值(PVf)			
	+(m+4)	手动操作值(MV <sub>MAN</sub> )	写入		
	+(m+5)	手动/自动选择(MAN/AUTO)			
	+(m+6)	报警(ALARM)	读取/写入		
	+(m+7) 至 +(m+17)	环路n的工作区 (不可使用)	读取/写入 禁止		

$m = (n-1) \times 18 + 10$

1) 应使用下面的公式计算设置 I/O 数据时使用的软元件点数。

$$\text{软元件点数} = 10 + 18 \times n \quad (n: \text{使用的环路数量})$$

2) 应以二进制值设置各个数据。

- 3) 初次处理标志是对 PID 运算开始时的处理方法进行设置。
- 在初次 PID 运算处理时，被视为已到达所设置的采样周期后进行运算。
  - 对初次处理标志进行以下设置。
    - 0 ..... 使用的环路数量的 PID 运算在一次扫描中被成批处理。
    - 0 以外 ..... 使用的环路数量的 PID 运算被分割在多次扫描中处理。  
从完成初次处理的环路开始依次进行采样。  
每次扫描中处理环路的数量是所设置的一次扫描执行的环路数量。
- 4) 对于 I/O 数据区中被指定为“写入”的数据应由用户通过顺控程序进行写入。  
对于 I/O 数据区中被指定为“读取”的数据应由用户通过顺控程序读取后使用。  
对于被指定为“读取/写入禁止”或“读取”的数据绝对不要进行写入。  
否则将不能正常地进行 PID 运算。  
但是，当要从初始状态开始控制时，必须通过顺控程序进行数据清除。
- 5) 如果使用的软元件点数超出指定软元件的最后软元件号，将会出错且不执行处理。(出错代码：4101)





### 6. PID 控制指令

PID 控制指令的指令构成与基本型 QCPU、高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU 和 QnACPU 的公共指令相同。

有关指令构成的详细内容，请参阅所使用的 CPU 模块的编程手册（公共指令篇）。



7. 怎样读指令说明

本章以后的指令说明将按以下形式进行。

8. 不完全微分 PID 控制指令和程序示例 MELSEC-Q/QnA

适用CPU	QCPU				QnA	Q4AR
	PLC CPU		过程CPU	冗余CPU		
	基本 △*	高性能 △*				
		×	○	×	×	

\*1:序列号的高5位为04122或以后  
\*2:序列号的高5位为05032或以后

1) 8.1.2 PID 运算

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[·]\[·]*3		特殊功能模块 U[·]\[G[·]	变址寄存器 Zn	常数	其它
	位	字		位	字				
⑤	—	○			—				

3) 指令符号 执行条件

4) [设置数据]

设置数据	内容	数据类型
⑤	分配到 I/O 数据区的软元件起始号	16 位二进制

5) [功能]

- 执行 S.PIDCONT 指令时，测定采样周期并执行 PID 运算。
- S.PIDCONT 指令以设置为用⑤指定的软元件号以后的 I/O 数据区的设置值(SV)和测定值(PV)为基础执行 PID 运算，且将运算结果存储到 I/O 数据区的自动操作值(MV)区中。
- 经过采样周期的设置时间后执行最初的 S.PIDCONT 指令时进行 PID 运算(参阅 5.1.2 节)。
- 在 PID 控制期间务必使控制指令 ON，且在每次扫描中执行 S.PIDCONT 指令。否则，将不能以正常的采样周期进行 PID 运算。此外，每次扫描中不能多次执行 S.PIDCONT 指令。如果在一次扫描中多次执行 S.PIDCONT 指令，将不能以正常的采样周期执行 PID 运算。
- 在中断程序、恒定扫描执行程序或低速执行程序不可使用 S.PIDCONT 指令。如果在中断程序、恒定扫描执行程序或低速执行程序中使用了 S.PIDCONT 指令，将不能以正常的采样周期执行 PID 运算。

8 - 3

- (1) 表示章节号和指令的简要说明。  
 (2) 对于指令中可使用的软元件附加○。  
 可使用软元件的使用种类划分如下所示：

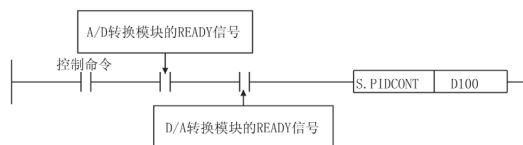
软元件分类	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[·]\[·]*3		特殊功能模块 U[·]\[G[·]	变址寄存器 Zn	常数*1	其它*1
	位	字		位	字				
可用软元件*4	X、Y、M、L、SM、F、B、SB、FX*2、FY*2	T*5、ST*5、C*5、D、W、SD、SW、FD	R、ZR	J[·]\X J[·]\Y J[·]\B J[·]\SB	J[·]\W J[·]\SW	U[·]\G[·]	Z	十进制数 十六进制数 实数常数 字符串常数	P、I、J、U、DX、DY、N、BL、TR、BLY\S

\*1: 可以设置的软元件记载在“常数”和“其它”一栏中。  
 \*2: FX 和 FY 只能与位数据一起使用，FD 只能与字数据一起使用。  
 \*3: 可以在 MELSECNET/G、MELSECNET/H 和 MELSECNET/10 中使用。  
 \*4: 关于相应软元件的说明，请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)或 QnACPU 编程手册(基础篇)。  
 \*5: T、ST 和 C 只能用于字软元件。

8. 不完全微分 PID 控制指令和程序示例

MELSEC-Q/QnA

- (6) 用  $\textcircled{S}$  指定在 I/O 数据区中指定的软元件号的起始号。如果 I/O 数据区，不要对文件寄存器 (R) 进行存储器保护。  
如果进行了存储器保护，虽然不会出错，但也不能正常地进行 PID 运算。  
有关 I/O 数据区的详细内容，请参阅 5.2 节。
- (7) 手动控制模式下，即使在输出手动操作值 (MV<sub>MAN</sub>) 时，在每次扫描中也应执行 S.PIDCONT 指令。  
如果未执行 S.PIDCONT 指令，则不能执行无冲击切换。  
有关无冲击切换的详细内容，请参阅 4.3.1 节。
- (8) 为了实现只有在读取 PV (测定值) 的 A/D 转换模块以及进行 MV (操作值) 输出的 D/A 转换模块正常时才能执行 S.PIDCONT 指令，应通过各个模块的 READY 信号进行互锁。\*



如果在各模块故障时执行 S.PIDCONT 指令，因为不能正确地读取 PV (测定值) 或不能正确地输出 MV (操作值)，所以不能正确地执行 PID 运算。

7) → [运算错误]

- (1) 在下列情况下，会发生运算错误，错误标志 (SM0) 将 ON，出错代码将被存储到 SD0 中。
  - 在执行 S.PIDCONT 指令之前执行了 S.PIDINIT 指令时。 (出错代码: 4103)
  - 在 I/O 数据区中设置的数据值超出允许范围时 (出错代码: 4100)
  - 当用  $\textcircled{S}$  指定的分配到 I/O 数据区的软元件范围超出相应软元件的最后一个软元件号时。 (出错代码: 4101)

备注

\*: 有关 A/D 转换模块以及 D/A 转换模块的 READY 信号的详细内容，请参阅相关模块的手册。

(3) 梯形图模式下的表示方式和指令执行条件如下所示:

执行条件	在 ON 的过程中执行	在 OFF→ON 时执行一次
在说明页使用的符号		

(4) 各个指令的设置数据的内容和数据类型如下所示:

数据类型	内容
16 位二进制	表示二进制 16 位数据或可以使用的字软元件的起始编号。

- (5) 表示指令的功能。
- (6) 表示指令能否在该 CPU 模块中使用。  
○: 可以使用; △: 在某种条件下可以使用; ×: 不可使用
- (7) 表示导致出错的条件和出错代码。

### 8. 不完全微分 PID 控制指令和程序示例

本章介绍用于执行 PID 控制的 PID 控制指令的使用方法及程序示例。

#### 8.1 PID 控制指令

## 8. 不完全微分 PID 控制指令和程序示例

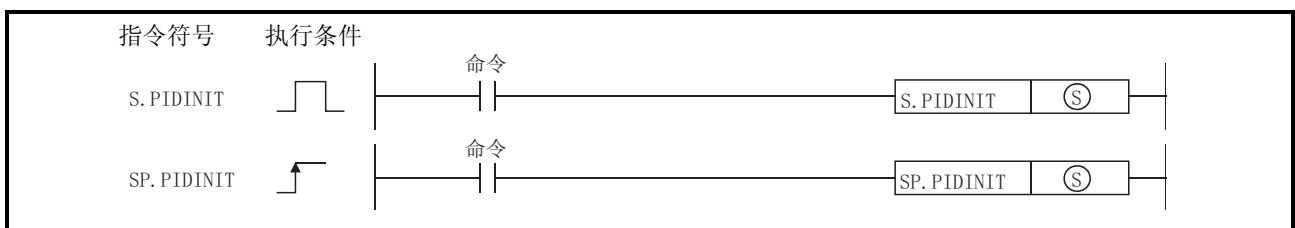
适用CPU	QCPU						LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU	冗余CPU				
	基本	高性能	通用						
△*1	△*2	○	×	○	○	×	×		

\*1: 序列号的高5位为04122或以后

\*2: 序列号的高5位为05032或以后

### 8.1.1 PID 控制用数据的设置

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[ ]\L[ ]		特殊功能模块 U[ ]\G[ ]	变址寄存器 Zn	常数	其它
	位	字		位	字				
⑤	-	○							



[设置数据]

设置数据	内容	数据类型
⑤	被设置为 PID 控制用数据的软元件的起始号	16 位二进制

[功能]

- (1) 可以将设置在⑤中指定的软元件号以后的使用环路数量的 PID 控制用数据批量地登录到 CPU 模块后，进行 PID 控制。  
有关 PID 控制用数据的详细内容，请参阅 5.1 节。
- (2) 在一次扫描中如果同时有多个 S.PIDINIT 指令被执行时，则在 S.PIDINIT 指令之前最后执行的 S.PIDINIT 指令的设置值有效。
- (3) 在执行 S.PIDCONT 指令之前必须执行 S.PIDINIT 指令。  
如果未执行 S.PIDINIT 指令则不能进行 PID 控制。

8

[运算错误]

- (1) 在下列情况下，会发生运算错误，出错标志 (SM0) 将 ON，出错代码将被存储在 SD0 中。
  - 设置为 PID 控制数据的值超出允许值范围时。 (出错代码: 4100)
  - (MV 上限) < (MV 下限) 时。 (出错代码: 4100)
  - (使用的环路数) < (一次扫描中执行的环路数) 时。 (出错代码: 4100)
  - 当⑤中指定的分配到 PID 控制用数据区的软元件范围超出相应软元件的最后一个软元件号时。 (出错代码: 4101)
  - 当在 5.1 节 (3) 中提到的 PID 控制用数据的“\*”区不是 0 时。 (出错代码: 4100)

## 8. 不完全微分 PID 控制指令和程序示例

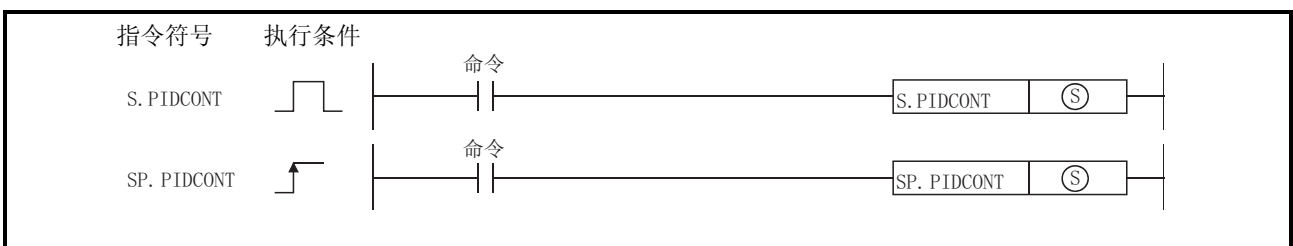
适用CPU	QCPU						LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU	冗余CPU				
	基本	高性能	通用						
	△*1	△*2	○	×	○	○	×	×	

\*1:序列号的高5位为04122或以后

\*2:序列号的高5位为05032或以后

### 8.1.2 PID 运算

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10(H) 直接 J[ ]\L[ ]		特殊功能模块 U[ ]\G[ ]	变址寄存器 Zn	常数	其它
	位	字		位	字				
⑤	-	○							



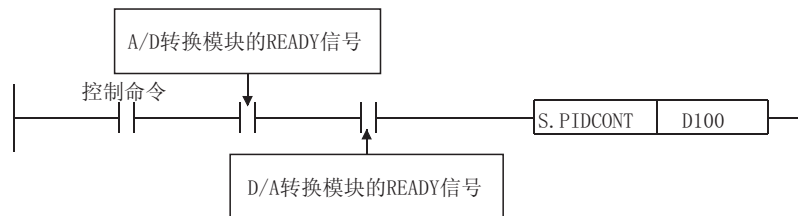
#### [设置数据]

设置数据	内容	数据类型
⑤	分配到 I/O 数据区的软元件起始号	16 位二进制

#### [功能]

- (1) 执行 S. PIDCONT 指令时，测定采样周期并执行 PID 运算。
- (2) S. PIDCONT 指令以设置在⑤中指定的软件号以后的 I/O 数据区的设置值(SV)和测定值(PV)为基础执行 PID 运算，且将运算结果存储到 I/O 数据区的自动操作值(MV)区中。
- (3) 经过采样周期的设置时间后执行最初的 S. PIDCONT 指令时进行 PID 运算(参阅 5.1.2 项)。
- (4) 在 PID 控制期间务必使控制指令 ON，且在每次扫描中执行 S. PIDCONT 指令。否则，将不能以正确的采样周期进行 PID 运算。此外，每次扫描中不能多次执行 S. PIDCONT 指令。如果在一次扫描中多次执行 S. PIDCONT 指令，将不能以正常的采样周期执行 PID 运算。
- (5) 在中断程序、恒定扫描执行型程序或低速执行型程序中不可使用 S. PIDCONT 指令。如果在中断程序、恒定扫描执行型程序或低速执行型程序中使用了 S. PIDCONT 指令，将不能以正常的采样周期执行 PID 运算。

- (6) 用⑤中指定 I/O 数据区中指定的软元件号的起始号。如果将文件寄存器(R)指定为 I/O 数据区，不要对文件寄存器(R)进行存储器保护。  
如果进行了存储器保护，虽然不会出错，但也不能正常地进行 PID 运算。  
有关 I/O 数据区的详细内容，请参阅 5.2 节。
- (7) 手动控制模式下，即使在输出手动操作值(MV<sub>MAN</sub>)时，在每次扫描中也应执行 S.PIDCONT 指令。  
如果未执行 S.PIDCONT 指令，则不能执行无冲击切换。  
有关无冲击切换的详细内容，请参阅 4.3.1 项。
- (8) 为了实现只有在读取 PV(测定值)的 A/D 转换模块以及进行 MV(操作值)输出的 D/A 转换模块正常时才能执行 S.PIDCONT 指令，应通过各个模块的 READY 信号进行互锁。\*



如果在各模块故障时执行 S.PIDCONT 指令，因为不能正确地读取 PV(测定值)或不能正确地输出 MV(操作值)，所以不能正确地执行 PID 运算。

### [运算错误]

- (1) 在下列情况下，会发生运算错误，出错标志(SM0)将 ON，出错代码将被存储到 SDO 中。
- 在执行 S.PIDCONT 指令之前未执行 S.PIDINIT 指令时。 (出错代码: 4103)
  - 在 PID 控制数据区中设置的数据值超出允许范围时。 (出错代码: 4100)
  - 当用⑤指定的分配到 PID 控制数据区域的软元件范围超出相应软元件的最后一个软元件号时。 (出错代码: 4101)

### 备注

\*: 有关 A/D 转换模块以及 D/A 转换模块的 READY 信号的详细内容，请参阅相关模块的手册。



## 8. 不完全微分 PID 控制指令和程序示例

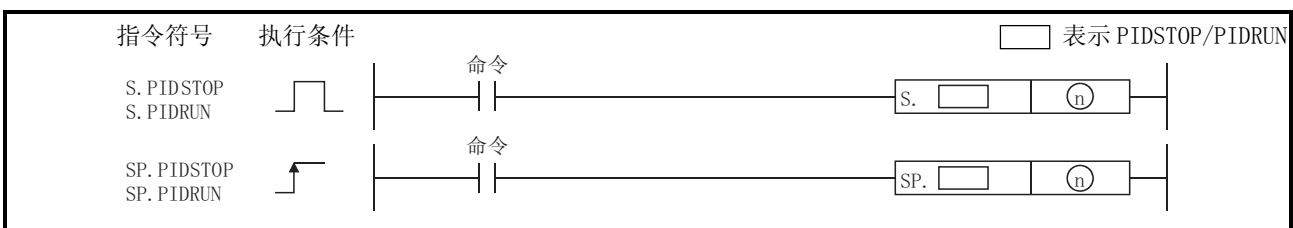
适用CPU	QCPU						LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU	冗余CPU				
	基本	高性能	通用						
	△*1	△*2	○	×	○	○	×	×	

\*1: 序列号的高5位为04122或以后

\*2: 序列号的高5位为05032或以后

### 8.1.3 停止/开始指定环路号的运算

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J: \: L: :		特殊功能模块 U: \: G: :	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑩	○		-		-		○	-	



#### [设置数据]

设置数据	内容	数据类型
⑩	要执行开始/停止的环路号	16 位二进制

#### [功能]

- (1) S. PIDSTOP、 SP. PIDSTOP
  - (a) 停止用⑩指定的环路号的 PID 运算。  
对于用 S. PIDSTOP 指令停止的环路，即使执行 S. PIDINIT 指令也不能再次开始 PID 运算。
  - (b) 停止期间保持运算数据。
- (2) S. PIDRUN、 SP. PIDRUN
  - (a) 开始用⑩指定的环路号的 PID 运算。  
该指令是用于重新执行被 S. PIDSTOP 指令停止的环路号的 PID 运算。
  - (b) 如果对正在执行 PID 运算的环路号执行了该指令，则该指令将被忽略。

#### [运算错误]

- (1) 在下列情况下，会发生运算错误且出错标志 (SM0) 将 ON，出错代码将被存储到 SD0 中。
  - 用⑩指定的环路号不存在时。 (出错代码: 4100)
  - 当⑩超出 1 到 8 的范围时。(基本型 QCPU) (出错代码: 4100)
  - 当⑩超出 1 至 32 的范围时。(高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU) (出错代码: 4100)
  - 当执行 S. PIDSTOP 指令之前未执行 S. PIDINIT 指令和 S. PIDCONT 指令时。 (出错代码: 4103)
  - 当执行 S. PIDRUN 指令之前未执行 S. PIDINIT 指令和 S. PIDCONT 指令时。 (出错代码: 4103)

## 8. 不完全微分 PID 控制指令和程序示例

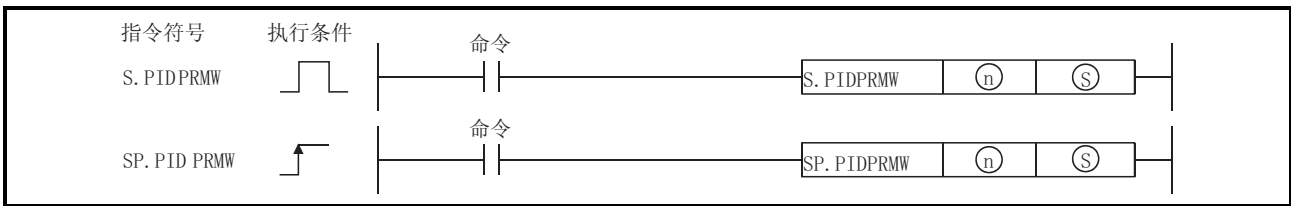
适用CPU	QCPU						LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU	冗余CPU				
	基本	高性能	通用						
	△*1	△*2	○	×	○	○	×	×	

\*1: 序列号的高5位为04122或以后

\*2: 序列号的高5位为05032或以后

### 8.1.4 指定环路的参数变更

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[ ]\G[ ]		特殊功能模块 U[ ]\G[ ]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①	○	○			-		○	-	
②	-	○			-		-	-	



#### [设置数据]

设置数据	内容	数据类型
①	要更改参数的环路号	16 位二进制
②	存储要更改的 PID 控制用数据的软元件的起始号	

#### [功能]

- 将用①指定的环路号的运算参数更改为存储于用②指定的起始软元件号以后的 PID 控制数据。
- 用②指定的软元件号以后的 PID 控制用数据的构成如下所示。有关 PID 控制用数据的详细内容，请参阅 5.1 节。

② +0	选择运算表达式
② +1	采样周期 (Ts)
② +2	比例常数 (Kp)
② +3	积分常数 (Ti)
② +4	微分常数 (Td)
② +5	过滤器系数 ( $\alpha$ )
② +6	MV 下限 (MVLL)
② +7	MV 上限 (MVHL)
② +8	MV 变化率限制 ( $\Delta$ MVL)
② +9	PV 变化率限制 ( $\Delta$ PVL)
② +10	0
② +11	微分增益 (Kd)
② +12	0
② +13	0

#### [运算错误]

- 在下列情况下，会发生运算错误，出错标志 (SM0) 将 ON，出错代码将被存储在 SDO 中。
  - 用①指定的环路号不存在时。 (出错代码: 4100)
  - 当①超出 1 至 8 的范围时。(基本型 QCPU) (出错代码: 4100)
  - 当①超出 1 至 32 的范围时。(高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU) (出错代码: 4100)
  - PID 控制用数据超出了设置范围时。 (出错代码: 4100)
  - PID 控制数据中②+10、②+12 和②+13 中任何一个不为 0 时。 (出错代码: 4100)
  - 用②指定的分配到 PID 控制用数据区的软元件范围超出相应软元件的最后一个软元件号时。 (出错代码: 4101)
  - 当执行 S. PIDPRMW 指令之前未执行 S. PIDINIT 指令时。 (出错代码: 4100)

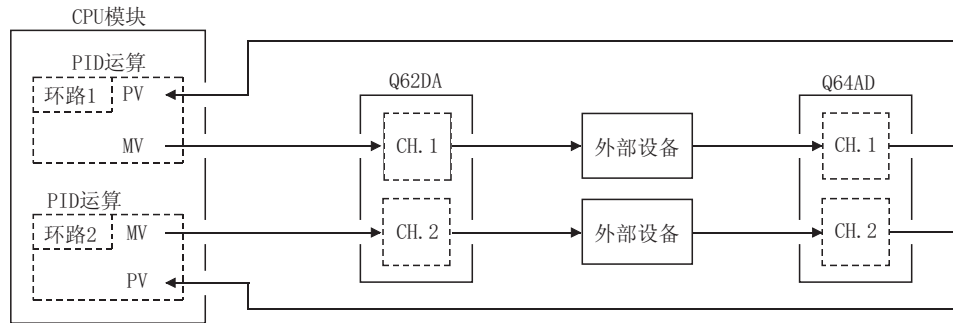
## 8. 不完全微分 PID 控制指令和程序示例

### 8.2 PID 控制程序示例

本节介绍执行 PID 控制的顺控程序的示例。

#### 8.2.1 程序示例的系统配置

下面阐述了在 8.2.2 项和 8.2.3 项中程序示例的系统配置。



Q64AD 的 I/O 号 ..... X/Y80 至 X/Y8F

Q62DA 的 I/O 号 ..... X/YA0 至 X/YAF

### 8.2.2 自动模式 PID 控制的程序示例

将从 Q64AD 获取的数字值作为 PV 值执行 PID 运算，将通过 PID 运算求出的 MV 值通过 Q64AD 输出，对外部设备进行控制的程序示例如下所示。

#### [程序条件]

- (1) 有关系统配置的详细内容，请参阅 8.2.1 项。
- (2) 执行 PID 运算的环路数为 2。
- (3) 采样周期为 1 秒。
- (4) 将 PID 控制用数据设置到下列软元件中：<sup>\*1</sup>
  - 公共数据 ..... D500 和 D501
  - 环路 1 用数据..... D502 至 D515
  - 环路 2 用数据..... D516 至 D529
- (5) 将 I/O 数据设置到下列软元件中：<sup>\*2</sup>
  - 公共数据 ..... D600 至 D609
  - 环路 1 用数据..... D610 至 D632
  - 环路 2 用数据..... D633 至 D655
- (6) 在顺控程序中将环路 1 和环路 2 的 SV 值设置为以下的值：
  - 环路 1 ..... 600
  - 环路 2 ..... 1000
- (7) 将下列软元件用作 PID 控制开始/停止指令。
  - PID 控制开始指令 ..... X0
  - PID 控制停止指令 ..... X1
- (8) 在 0 至 2000 的范围之内设置 Q64AD 和 Q62DA 的数字值。

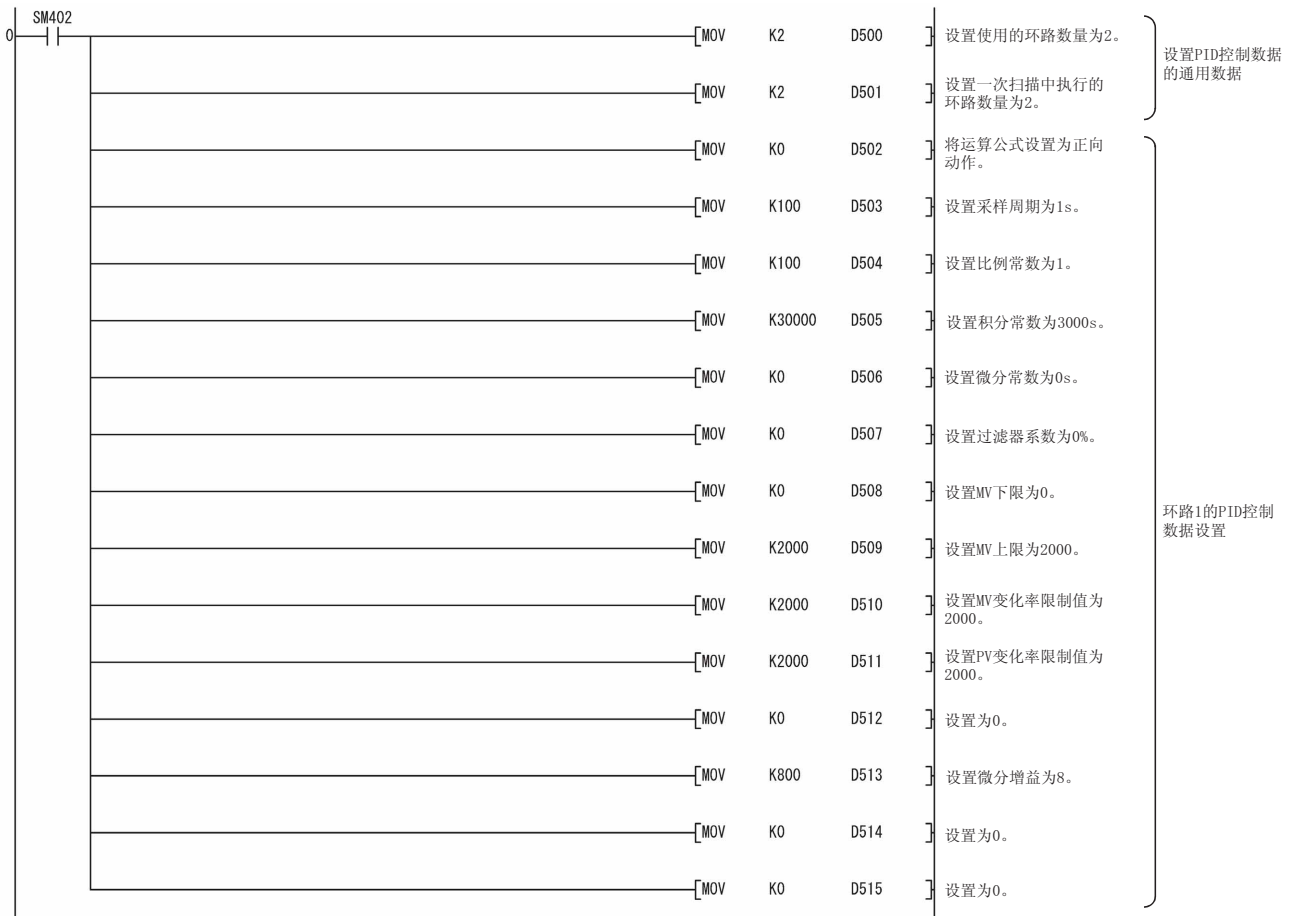
备注
----

\*1: 有关 PID 控制用数据的详细内容，请参阅 5.1 节。

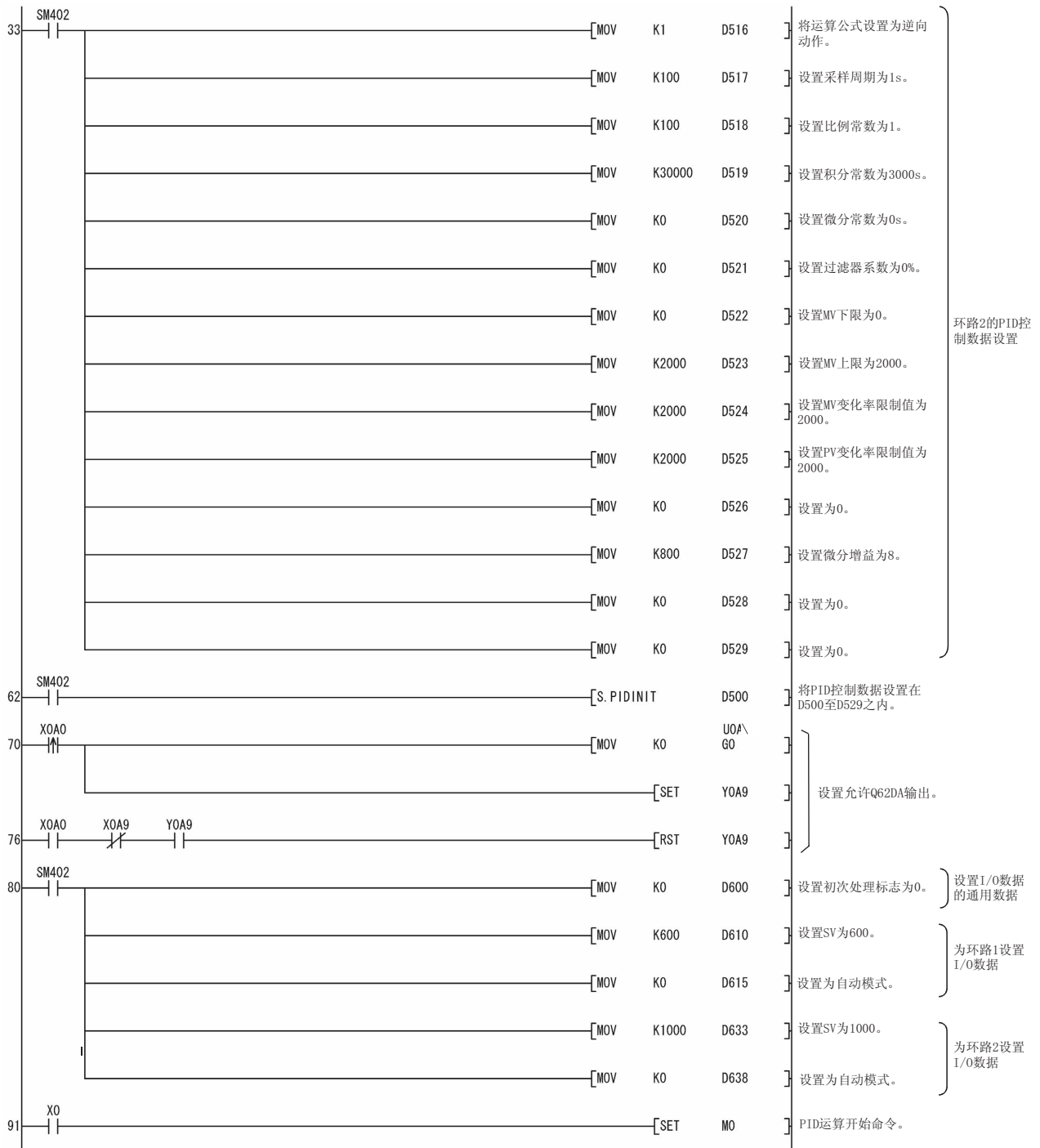
\*2: 有关 I/O 数据的详细内容，请参阅 5.2 节。

## 8. 不完全微分 PID 控制指令和程序示例

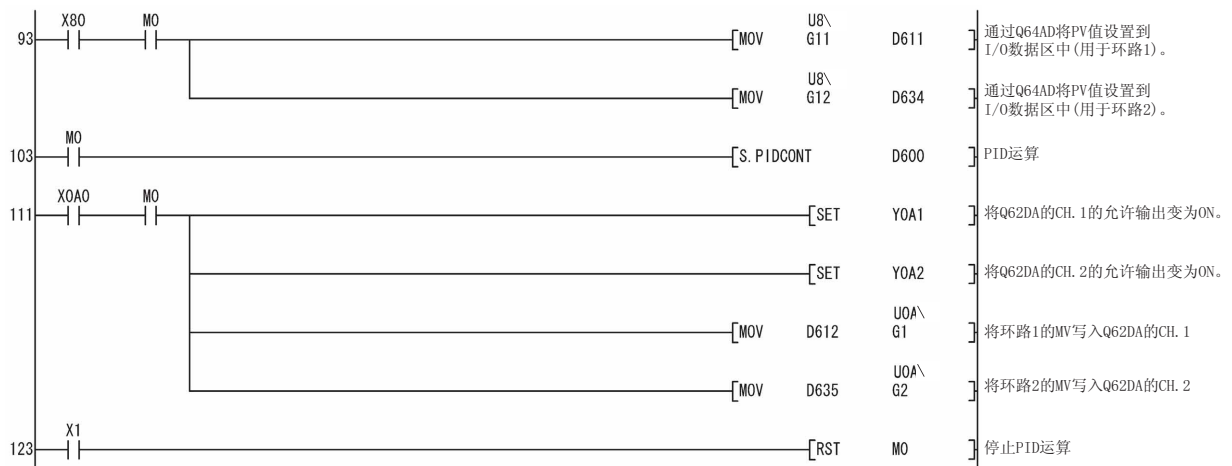
### [程序示例]



## 8. 不完全微分 PID 控制指令和程序示例



## 8. 不完全微分 PID 控制指令和程序示例

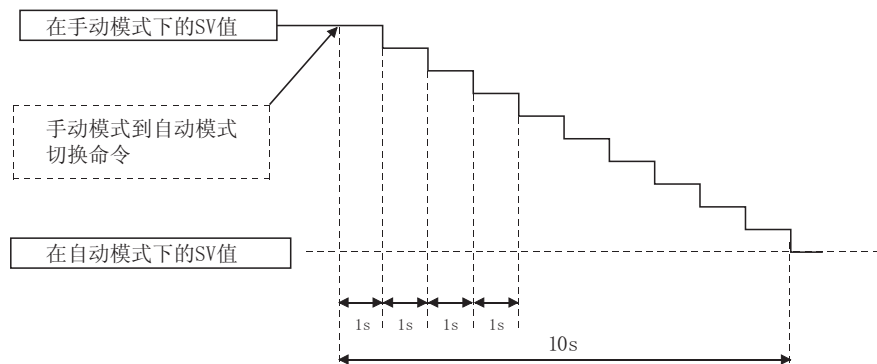


8.2.3 在自动和手动间切换 PID 控制模式的程序示例

下面介绍执行 PID 运算时在自动模式和手动模式之间切换的程序示例。

[程序条件]

- (1) 有关系统配置的详细内容，请参阅 8.2.1 项。
- (2) 执行 PID 运算的环路数为 1。
- (3) 采样周期为 1s。
- (4) 在下列软元件中设置 PID 控制用数据。  
 公共数据 ..... D500 和 D501  
 环路 1 用数据..... D502 至 D515
- (5) 在下列软元件中设置 I/O 数据。  
 公共数据 ..... D600 至 D609  
 环路 1 用数据..... D610 至 D630
- (6) 用外部数字开关设置 SV 和手动模式时的 MV。  
 SV ..... X30 至 X3F  
 MV(手动控制模式) ..... X20 至 X2F
- (7) 在 PID 控制的开始/停止及自动/手动切换指令中使用下列软元件。  
 PID 控制开始指令 ..... X0  
 PID 控制停止指令 ..... X1  
 SV 设置指令 ..... X3  
 手动模式下的 MV 设置指令..... X4  
 自动/手动模式切换指令 ..... X6  
 (OFF: 自动模式; ON: 手动模式)
- (8) 在 0 至 2000 的范围之内设置 Q64AD 和 Q62DA 的数字值。
- (9) 将 PID 无冲击处理标志 SM794 置于 OFF。在手动模式下，当执行 PID 运算时，SV 被自动地改写成 PV。因此，当把手动模式重新切换到自动模式时，应将 SV 值改写为在切换到手动模式之前自动模式控制时使用的 SV 值。  
 SV 值的改写应按下图所示将 SV 值分 10 阶段逐步进行改写。



使用如下所示的运算方法改写 SV。

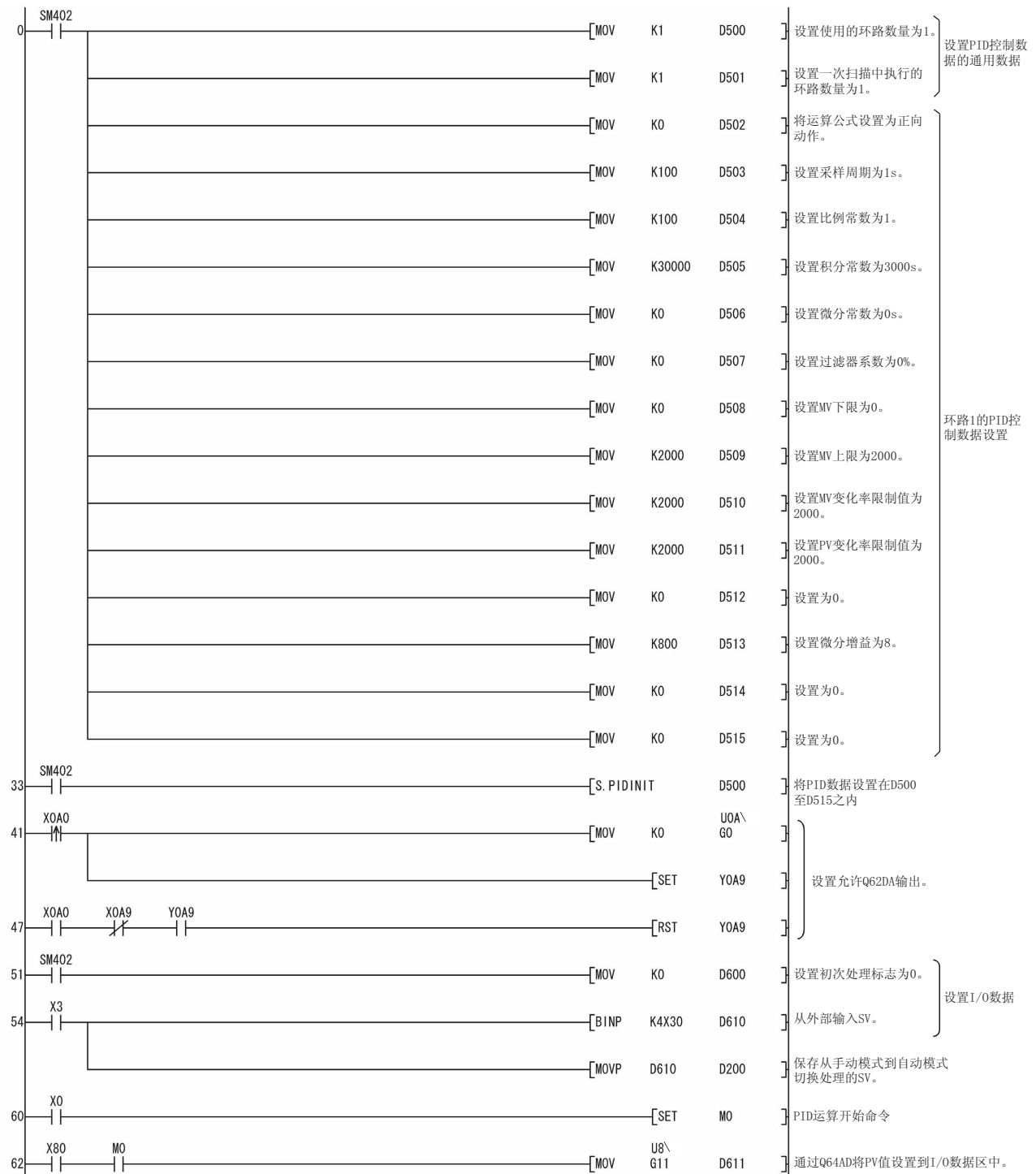
$$\frac{\left( \text{在手动模式下的SV值} \right) - \left( \text{在自动模式下的SV值} \right)}{10} = \boxed{\text{减法值}} \cdots \cdots \boxed{\text{余数}}$$

进行 SV 值的减法运算时，以上式的“减法值”逐秒进行减法运算。在进行第一次减法运算的同时加上余数。

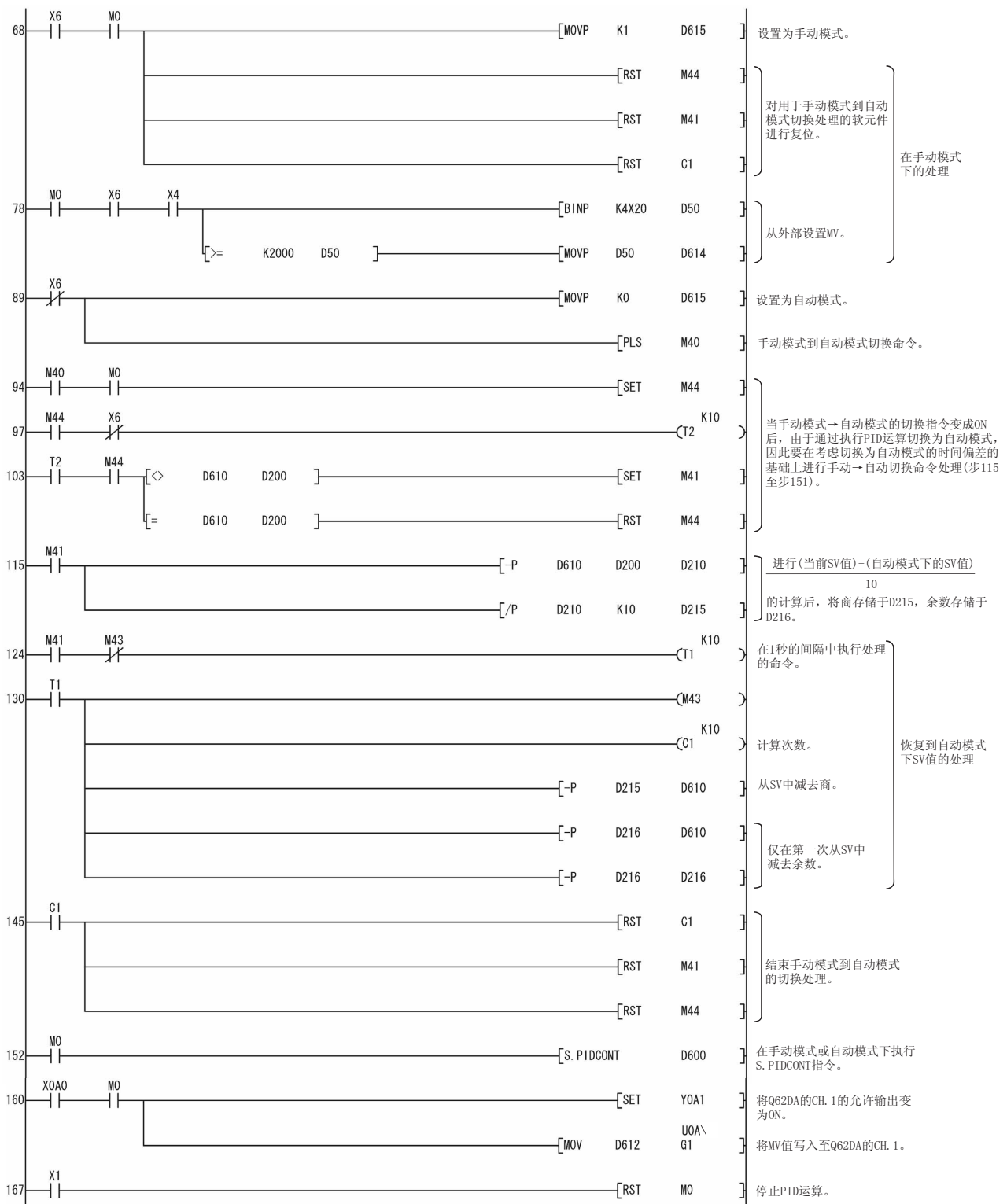


## 8. 不完全微分 PID 控制指令和程序示例

### [程序示例]



## 8. 不完全微分 PID 控制指令和程序示例



### 9. 完全微分 PID 控制指令和程序示例

本章介绍 PID 控制指令的使用方法以及执行 PID 控制的程序示例。

#### 9.1 PID 控制指令

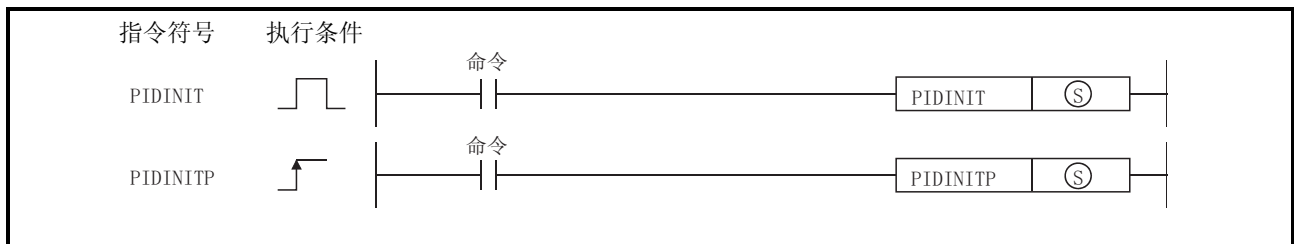
## 9. 完全微分 PID 控制指令和程序示例

适用CPU	QCPU					LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU	冗余CPU			
	基本	高性能	通用					
	△*1	○	○	×	○	○	○	

\*1: 序列号的高5位为04122或以后

### 9.1.1 PID 控制数据的设置

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[ ]\G[ ]		特殊功能模块 U[ ]\G[ ]	变址寄存器 Zn	常数	其它
	位	字		位	字				
⑤	-	○					-		



#### [设置数据]

设置数据	内容	数据类型
⑤	被设置了PID控制用数据的软元件的起始号	16位二进制

#### [功能]

- 将设置在⑤中指定的软元件号以后的使用环路数量的PID控制用数据批量地登录至CPU模块中，从而可以进行PID控制。  
有关PID控制用数据的详细内容，请参阅5.1节。
- 在一次扫描中多次执行了PIDINIT指令时，则在PIDCONT指令之前最近执行的PIDINIT指令的设置值有效。
- 在执行PIDCONT指令之前必须执行PIDINIT指令。  
如果未执行PIDINIT指令则不能进行PID控制。

#### [运算错误]

- 在下列情况下，会发生运算错误，出错标志(SM0)将ON，出错代码将被存储在SD0中。
  - 当设置为PID控制数据的值超出允许范围时。 (出错代码: 4100)
  - 当(使用的环路数)<(一次扫描中执行的环路数)时。 (出错代码: 4100)
  - 当(MV上限)<(MV下限)时。 (出错代码: 4100)
  - 当⑤中指定的分配到PID控制用数据区的软元件范围超出相应软元件的最后一个软元件号时。 (出错代码: 4101)

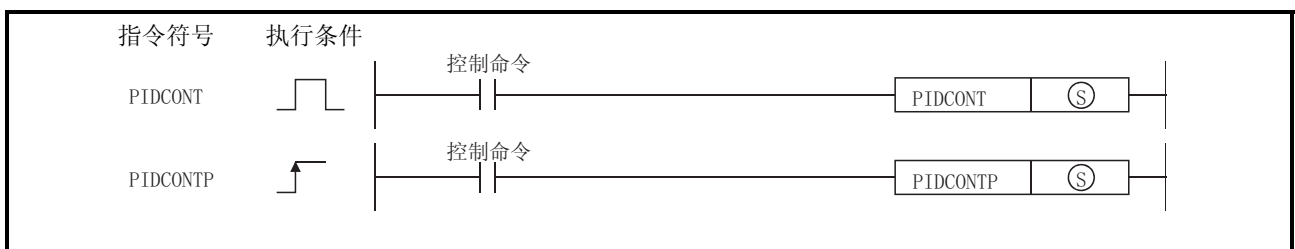
## 9. 完全微分 PID 控制指令和程序示例

适用CPU	QCPU				LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU			
	基本	高性能	通用				
	△*1	○	○	×	○	○	○

\*1:序列号的高5位为04122或以后

### 9.1.2 PID 控制

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[ ]\G[ ]		特殊功能模块 U[ ]\G[ ]	变址寄存器 Zn	常数	其它
	位	字		位	字				
⑤	-	○							



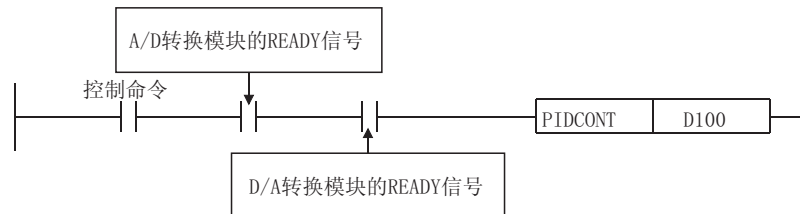
#### [设置数据]

设置数据	说明	数据类型
⑤	分配到 I/O 数据区的软元件起始号	16 位二进制

#### [功能]

- (1) 执行 PIDCONT 指令时，测定采样周期并执行 PID 运算。
- (2) PIDCONT 指令以设置在⑤中指定的软元件号以后的 I/O 数据区的设置值(SV)和测定值(PV)为基础执行 PID 运算，且将结果存储到 I/O 数据区的自动操作值(MV)区中。
- (3) 经过采样周期的设置时间后，执行最初的 PIDCONT 指令时进行 PID 运算(参阅 5.1.2 节)。
- (4) 在 PID 控制期间务必使控制指令 ON，且在每次扫描中执行 PIDCONT 指令。否则，将不能以正常的采样周期进行 PID 运算。每次扫描中只能执行一次 PIDCONT 指令。如果一次扫描中多次执行了 PIDCONT 指令，将不能以正常采样周期执行 PID 运算。
- (5) 在中断程序、恒定扫描执行型程序或低速执行型程序中不可使用 PIDCONT 指令。如果在中断程序、恒定扫描执行型程序或低速执行型程序中使用了 PIDCONT 指令，将不能以正常的采样周期执行 PID 运算。

- (6) 用⑤指定在 I/O 数据区中指定的软元件号的起始号。如果将文件寄存器(R)指定为 I/O 数据区，不要将文件寄存器(R)的存储器保护设置为 ON。  
如果将存储器保护设置为 ON，虽然不会出错，但也不能正常地进行 PID 运算。有关 I/O 数据区的详细内容，请参阅 5.2 节。
- (7) 即使在手动控制模式下输出手动操作值(MV<sub>MAN</sub>)时，在每次扫描中也应执行 PIDCONT 指令。  
如果未执行 PIDCONT 指令，则不能执行无冲击功能。  
有关无冲击功能的详细内容，请参阅 4.3.1 项。
- (8) 为了实现只有在读取 PV(测定值)的 A/D 转换模块以及进行 MV(操作值)输出的 D/A 转换模块正常时才能执行 PIDCONT 指令，应通过各个模块的 READY 信号进行互锁。\*



如果在各模块故障时执行 PIDCONT 指令，因为不能正确地读取 PV(测定值)或不能正确地输出 MV(操作值)，所以不能正确地执行 PID 运算。

[运算错误]

- (1) 在下列情况下，会发生运算错误，出错标志(SM0)将 ON，出错代码将被存储在此 SDO 中。
- 在执行 PIDCONT 指令之前未执行 PIDINIT 指令时。 (出错代码: 4103)
  - 在 PID 控制数据区设置的数据超出允许范围时。 (出错代码: 4100)
  - 当⑤中指定的分配到 PID 控制数据区域的软元件范围超出相应软元件的最后一个软元件号时。 (出错代码: 4101)

**备注**

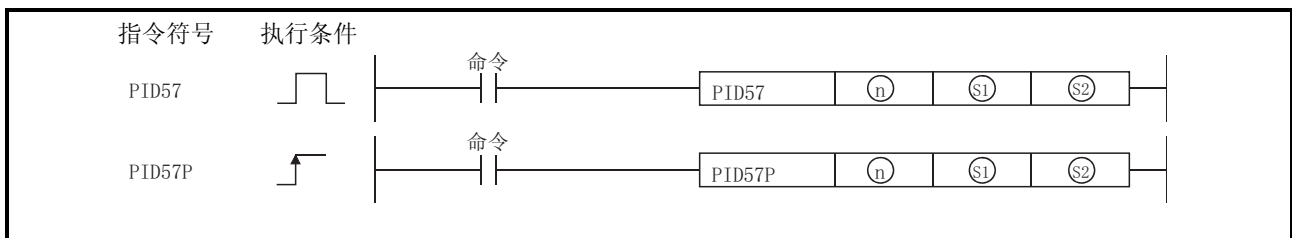
\*: 有关 A/D 转换模块和 D/A 转换模块的 READY 信号的详细内容，请参阅相关模块的手册。

## 9. 完全微分 PID 控制指令和程序示例

通用CPU	QCPU					LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU	冗余CPU			
	基本	高性能	通用					
	×	×	×	×	×	×	○	○

### 9.1.3 监视 PID 控制状态 (仅用于 QnACPU)

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[ ]\G[ ]		特殊功能模块 U[ ]\G[ ]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
①							○	-	
①							○	-	
②							-	-	



#### [设置数据]

设置数据	说明	数据类型
①	AD57 (S1) 中用于监视 PID 控制状态的起始 I/O 号	16 位二进制
①	与要进行监视的环路号相对应的画面号	
②	初始画面显示请求	

#### [功能]

- 用①中指定的 AD57 (S1) 的显示器以条形图显示①中指定的环路号的 PID 控制状态。  
在 PID 控制监视执行之初，通过执行②中指定的初始画面显示请求，进行监视画面 (条形图和数字值除外) 静止部分中的字符显示。
- AD57 (S1) 的 VDAM 区中地址 0 至 1599 被用于 PID 控制监视。  
因此，如果执行了 PID 控制状态监视，用户就不能使用这些地址。如果使用了这些地址，存储于其中的数据将会丢失。
- 在进行 PID 控制监视时，在执行 PID57 指令之前必须先执行 CMODE 指令 (AD57 命令)。  
如果未用 CMODE 指令将 AD57 (S1) 设置成 CRT 标准显示模式，则显示器将不能进行显示。
- 只有在执行了 PIDINIT 指令和 PIDCONT 指令之后才能执行 PID57 指令。  
如果在执行 PIDINIT 指令和 PIDCONT 指令之前执行 PID57 指令，则会发生错误。

(5) ①中指定的“1~4”的画面编号所对应的监视环路号如下所示。

画面号	监视的环路号
1	环路 1 至环路 8
2	环路 9 至环路 16
3	环路 17 至环路 24
4	环路 25 至环路 32

(6) ②中所指定的初始画面显示请求是用来指定显示监视画面中静止部分的字符。

执行初始画面显示请求时，将②设置为“0”。

在未执行初始画面显示请求时，不显示除条形图及数值数据以外的字符。

(7) 显示初始画面后，①中指定的值将被自动地存储到②中，然后执行 PID 控制监视功能。

如果②中指定的软元件是文件寄存器，不要将文件寄存器的存储器保护功能设置为 ON。

如果将文件寄存器的存储保护功能设置为 ON，画面则不能正确显示监视数据。

(8) QnACPU 运行后，应通过最初的 PID57 指令仅执行一次初始画面显示请求。

如果每次扫描都执行 PID57 指令，则只显示出静止部分的字符而不能显示条形图及数值数据。

(9) 若用 AD57 (S1) 监视 PID 控制状态，必须在 AD57 (S1) 中装载上字符发生器 ROM 及画布 ROM。

必须在字符发生器 ROM 中创建如图 9.1 所示的字符 (与字符代码 000 至 00Bh 相应)。

如果未创建这些字符，则不能显示条形图。

有关创建字符发生器 ROM 及画布 ROM 的详细内容，请参阅下列手册。

- SW1GP-AD57P 操作手册



## 9. 完全微分 PID 控制指令和程序示例

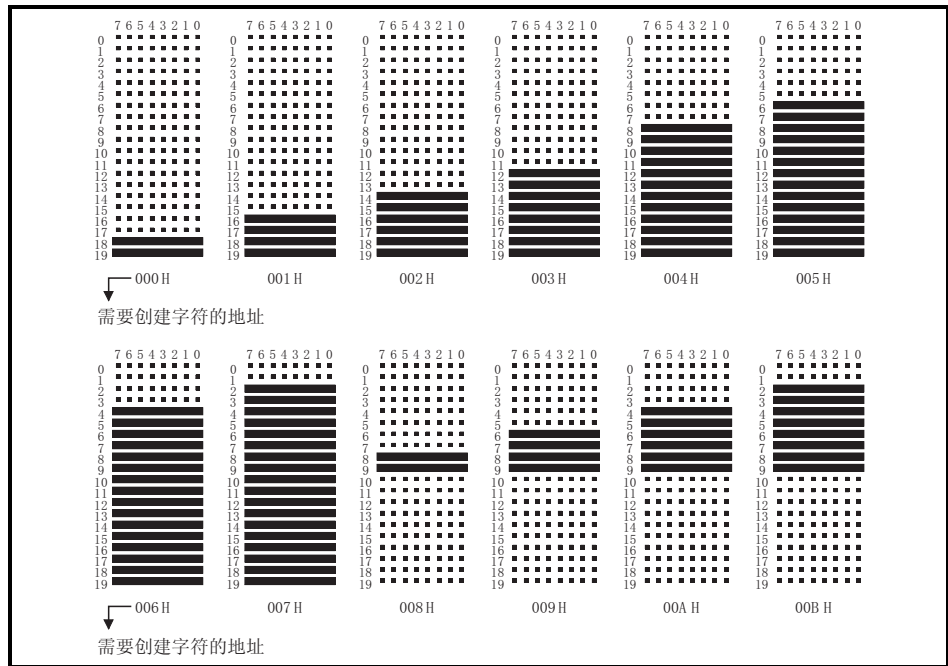


图 9.1 用于 PID 控制状态监视的字符

### [运算错误]

(1) 在下列情况下，会发生运算错误，出错标志 (SM0) 将 ON，出错代码将被存储在 SDO 中。

- 未对 AD57 (S1) 执行 CMODE 指令时。 (出错代码: 2110)
- 执行 PID57 指令之前未执行 PIDINIT 指令时。 (出错代码: 4103)
- 执行 PID57 指令之前未执行 PIDCONT 指令时。 (出错代码: 4103)
- ④中指定的画面号超出 1 至 4 的范围时。 (出错代码: 4100)

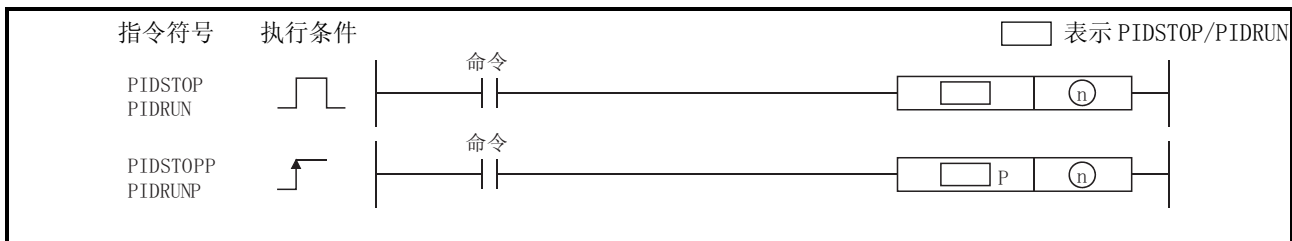
## 9. 完全微分 PID 控制指令和程序示例

适用CPU	QCPU					LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU	冗余CPU			
	基本	高性能	通用					
	△*1	○	○	×	○	○	○	

\*1: 序列号的高5位为04122或以后

### 9.1.4 停止/开始指定环路号的运算

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[ ]\G[ ]		特殊功能模块 U[ ]\G[ ]	变址寄存器 Zn	常数 K, H	其它
	位	字		位	字				
⑩	○							-	



#### [设置数据]

设置数据	说明	数据类型
⑩	要执行开始/停止的环路号	16 位二进制

#### [功能]

- (1) PIDSTOP、PIDSTOPP
  - (a) 停止用⑩中指定的环路号的 PID 运算。  
被 PIDSTOP 指令停止的环路即使执行了 PIDINIT 指令也不能重新开始 PID 运算。
  - (b) 停止期间保持运算数据。
- (2) PIDRUN、PIDRUNP
  - (a) 开始用⑩中指定的环路号的 PID 运算。  
该指令是用来重新执行被 PIDSTOP 指令停止的环路号的 PID 运算的指令。
  - (b) 如果对当前正在进行 PID 运算的环路号执行了该指令，则该指令将被忽略。

#### [运算错误]

- (1) 在下列情况下，会发生运算错误，且出错标志 (SM0) 将 ON。出错代码将被存储到 SDO 中。
  - ⑩中指定的环路号不存在时。 (出错代码: 4100)
  - ⑩超出 1 至 8 的范围时。(基本型 QCPU) (出错代码: 4100)
  - ⑩超出 1 至 32 的范围时。(高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU、QnACPU) (出错代码: 4100)
  - 执行 PIDSTOP 指令之前未执行 PIDINIT 指令和 PIDCONT 指令时。 (出错代码: 4103)
  - 执行 PIDRUN 指令之前未执行 PIDINIT 指令和 PIDCONT 指令时。 (出错代码: 4103)

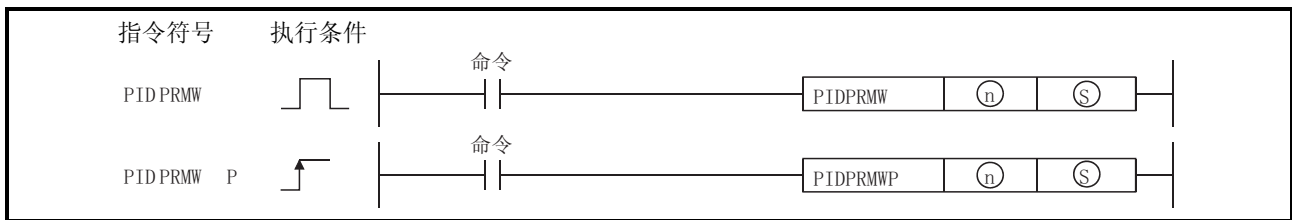
## 9. 完全微分 PID 控制指令和程序示例

适用CPU	QCPU					LCPU	QnA	Q4AR
	可编程控制器CPU			过程CPU	冗余CPU			
	基本	高性能	通用					
	△*1	○	○	×	○	○	○	

\*1: 序列号的高5位为04122或以后

### 9.1.5 更改指定环路号的参数

设置数据	可用软元件								
	内部软元件 (系统、用户)		文件寄存器	MELSECNET/10 (H) 直接 J[ ]\G[ ]		特殊功能模块 U[ ]\G[ ]	变址寄存器 Zn	常数 K、H	其它
	位	字		位	字				
①	○	○	○		○		-	-	
②	-	○	-		-		-	-	



#### [设置数据]

设置数据	说明	数据类型
①	要更改参数的环路号	16 位二进制
②	存储要进行更改的 PID 控制用数据的软元件起始号	

#### [功能]

- 将①中指定的环路号的运算参数更改为存储于②中指定的起始软元件号以后的 PID 控制数据。
- ②中指定的软元件号以后的 PID 控制用数据的构成如下所示。有关 PID 控制用数据的详细内容，请参阅 5.1 节。

② +0	选择运算公式
② +1	采样周期(Ts)
② +2	比例常数(Kp)
② +3	积分常数(Ti)
② +4	微分常数(Td)
② +5	过滤器常数(α)
② +6	MV下限(MVLL)
② +7	MV上限(MVHL)
② +8	MV变化率限制值(ΔMVL)
② +9	PV变化率限制值(ΔPVL)

#### [运算错误]

- 在下列情况下，会发生运算错误，出错标志(SM0)将 ON，出错代码将被存储在 SD0 中。
  - 用①中指定的环路号不存在时。 (出错代码：4100)
  - 当①超出 1 至 8 的范围时。(基本型 QCPU) (出错代码：4100)
  - 当①超出 1 至 32 的范围时。(高性能型 QCPU、冗余 CPU、通用型 QCPU、LCPU、QnACPU) (出错代码：4100)
  - PID 控制用数据超出允许范围时。 (出错代码：4100)
  - 当②中指定的分配到 PID 控制用数据区的软元件范围超出相应软元件的最后一个软元件号时。 (出错代码：4101)
  - 当执行 PIDPRMW 指令之前未执行 PIDINIT 指令时。 (出错代码：4103)

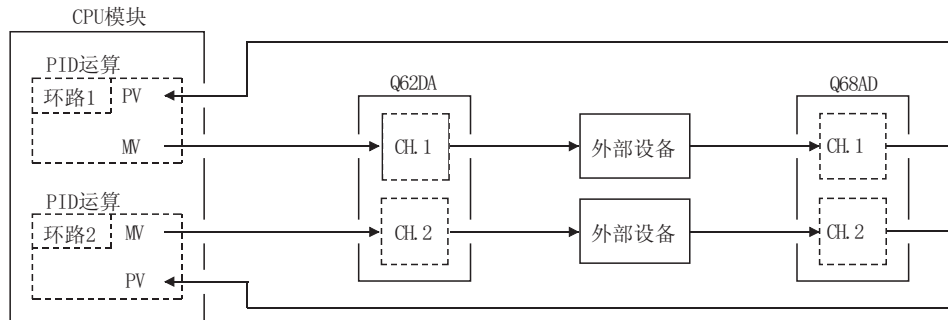
## 9. 完全微分 PID 控制指令和程序示例

### 9.2 PID 控制程序示例 (用于 QCPU、LCPU)

本节介绍执行 PID 控制的顺控程序示例。

#### 9.2.1 程序示例的系统配置

下面阐述了 9.2.2 项和 9.2.3 项中程序示例的系统配置。



Q64AD 的 I/O 号 ..... X/Y80 至 X/Y8F

Q62DA 的 I/O 号 ..... X/YA0 至 X/YAF

### 9.2.2 自动模式 PID 控制的程序示例

将从 Q64AD 获取的数字值作为 PV 值执行 PID 运算，将通过 PID 运算求出的 MV 值通过 Q62DA 输出，对外部设备进行控制的程序示例如下所示。

#### [程序条件]

- (1) 有关系统配置的详细内容，请参阅 9.2.1 项。
- (2) 执行 PID 运算的环路数为 2。
- (3) 采样周期为 1s。
- (4) 将 PID 控制用数据设置到下列软元件中。\*1  
公共数据..... D500 和 D501  
环路 1 用数据..... D502 至 D511  
环路 2 用数据..... D512 至 D521
- (5) 将 I/O 数据设置到下列软元件中。\*2  
公共数据..... D600 至 D609  
环路 1 用数据..... D610 至 D627  
环路 2 用数据..... D628 至 D645
- (6) 在顺控程序中将环路 1 和环路 2 的值设置为以下的值。  
环路 1 ..... 600  
环路 2 ..... 1000
- (7) 下列软元件用作 PID 控制开始/停止指令。  
PID 控制开始指令 ..... X0  
PID 控制停止指令 ..... X1
- (8) 在 0 至 2000 的范围之内设置 Q64AD 和 Q62DA 的数字值。

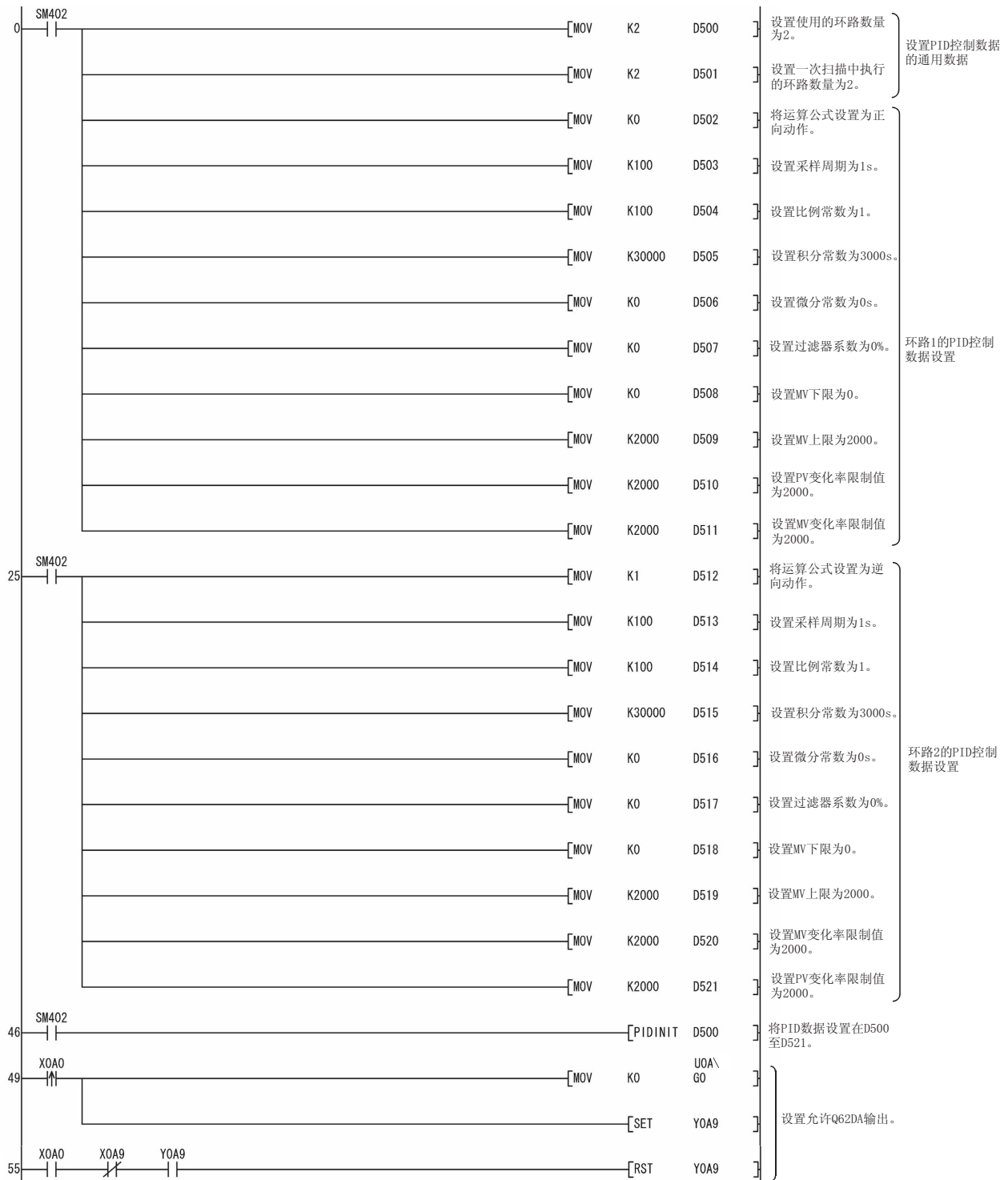
备注
----

\*1: 有关 PID 控制数据的详细内容，请参阅 5.1 节。

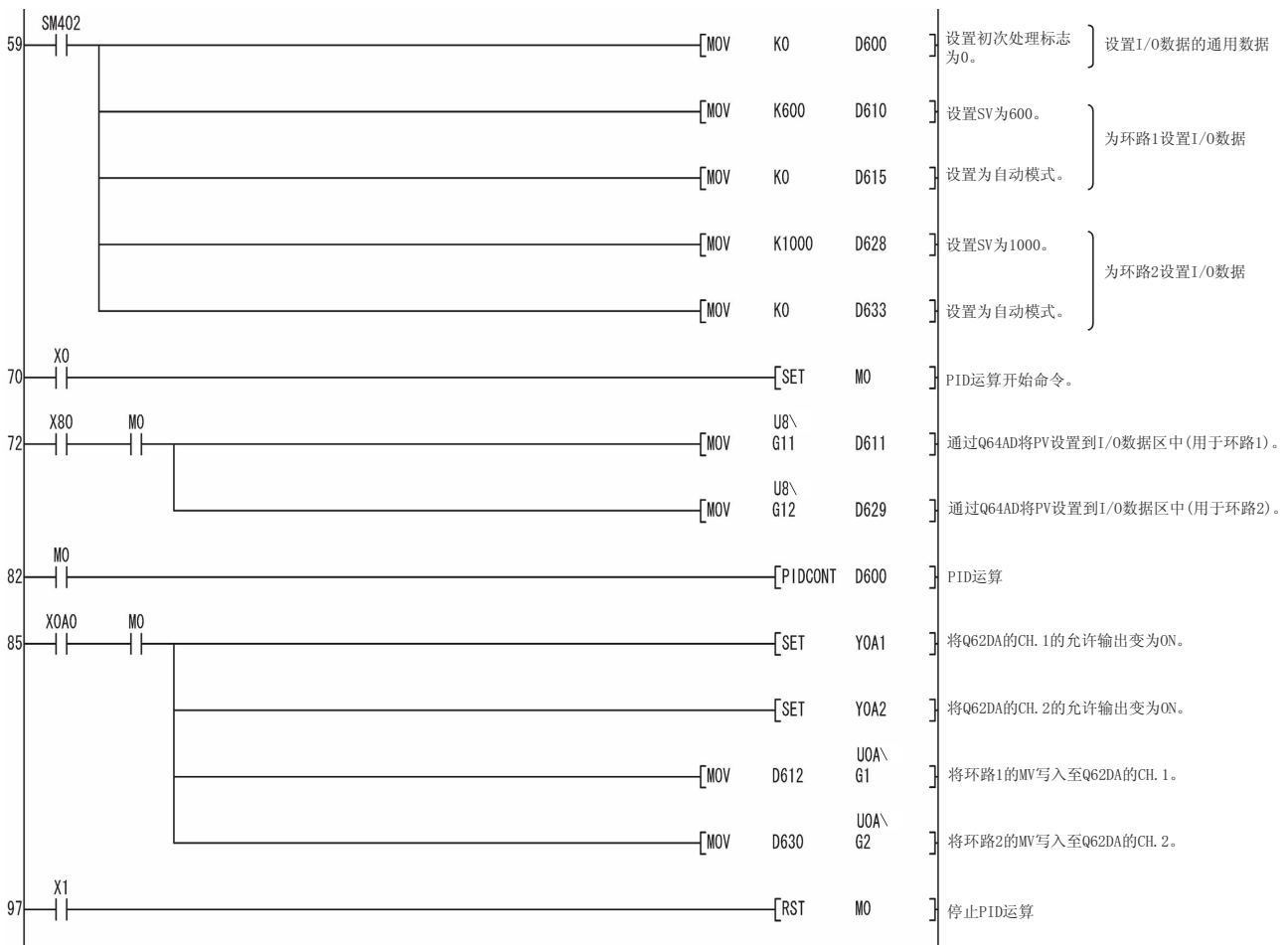
\*2: 有关 I/O 数据的详细内容，请参阅 5.2 节。

## 9. 完全微分 PID 控制指令和程序示例

### [程序示例]



## 9. 完全微分 PID 控制指令和程序示例



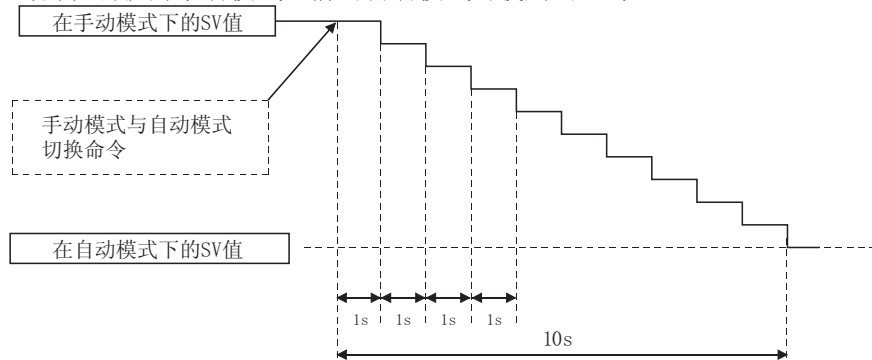
9.2.3 在自动和手动间切换 PID 控制模式的程序示例

下面介绍执行 PID 运算时在自动模式和手动模式间切换的程序示例。

[程序条件]

- (1) 有关系统配置的详细内容，请参阅 9.2.1 项。
- (2) 执行 PID 运算的环路数为 1。
- (3) 采样周期为 1s。
- (4) 在下列软元件中设置 PID 控制数据。
  - 公共数据..... D500 和 D501
  - 环路 1 用数据..... D502 至 D511
- (5) 在下列软元件中设置 I/O 数据。
  - 公共数据..... D600 至 D609
  - 环路 1 用数据..... D610 至 D627
- (6) 用外部数字开关设置 SV 和手动模式时的 MV。
  - SV..... X30 至 X3F
  - MV(手动控制模式时)..... X20 至 X2F
- (7) 在 PID 控制的开始/停止及自动/手动切换指令中使用下列软元件。
  - PID 控制开始指令..... X0
  - PID 控制停止指令..... X1
  - SV 设置指令..... X3
  - 手动模式下的 MV 设置指令..... X4
  - 自动/手动模式切换指令..... X6

(OFF:自动模式, ON:手动模式)
- (8) 在 0 至 2000 的范围之内设置 Q64AD 和 Q62DA 的数字值。
- (9) 将 PID 无冲击处理标志 SM774 置于 OFF。在手动模式下，当执行 PID 运算时 SV 将被自动地改写成 PV，因此，当把手动模式重新切换到自动模式时，应将 SV 改写为在切换为手动模式之前的自动模式时使用的 SV 值。



使用如下所示的运算方法改写 SV。

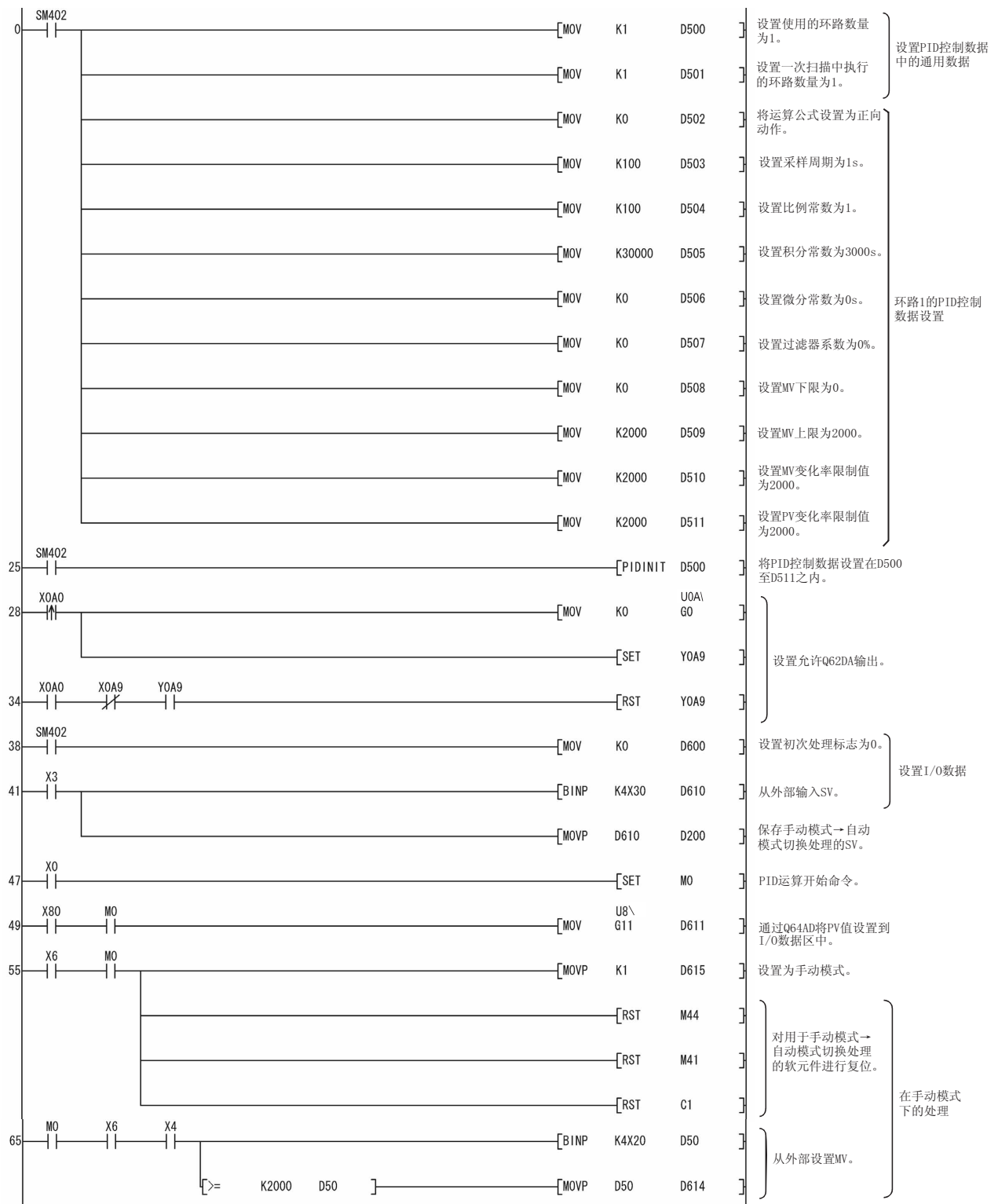
$$\frac{\left( \text{在手动模式下的SV值} \right) - \left( \text{在自动模式下的SV值} \right)}{10} = \boxed{\text{减数值}} \cdots \cdots \boxed{\text{余数}}$$

进行 SV 值的减法运算时，以上式的减数值逐秒进行减法运算。在进行第一次减法运算的同时减去余数。



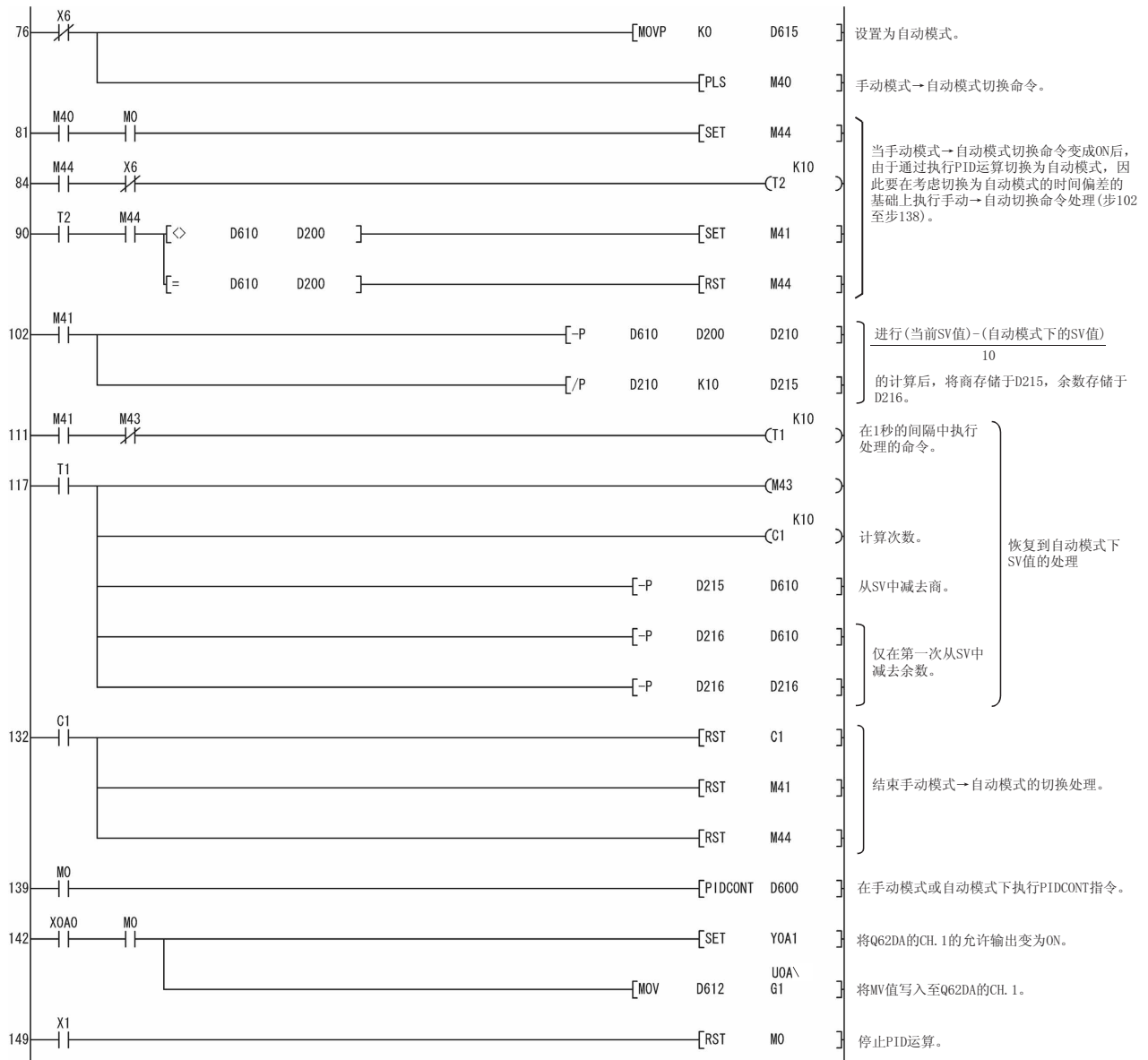
## 9. 完全微分 PID 控制指令和程序示例

### [程序示例]



## 9. 完全微分 PID 控制指令和程序示例

### [程序示例]



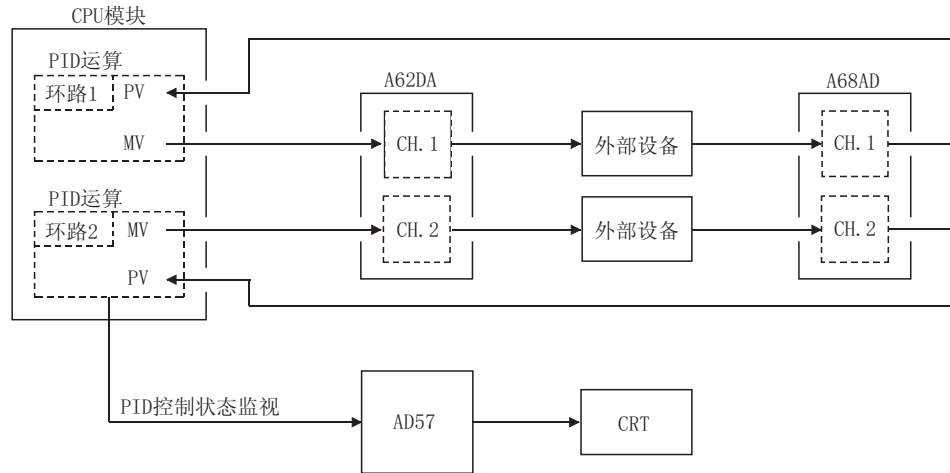
## 9. 完全微分 PID 控制指令和程序示例

### 9.3 PID 控制程序示例 (仅用于 QnACPU)

本节介绍执行 PID 控制的顺控程序的示例。

#### 9.3.1 程序示例的系统配置

下面阐述了在 9.3.2 项和 9.3.3 项中程序示例的系统配置。



A68AD 的 I/O 号 ..... X/Y80 至 X/Y9F

A62DA 的 I/O 号 ..... X/YA0 至 X/YBF

AD57 的 I/O 号 ..... X/YC0 至 X/YFF

### 9.3.2 自动模式 PID 控制的程序示例

将从 A68AD 获取的数字值作为 PV 值执行 PID 运算，将通过 PID 运算求出的 MV 值通过 A62DA 输出，对外部设备进行控制的程序示例如下所示。

#### [程序条件]

- (1) 有关系统配置的详细内容，请参阅 9.3.1 项。
- (2) 执行 PID 运算的环路数为 2。
- (3) 采样周期为 1s。
- (4) 将 PID 控制用数据设置到下列软元件中。\*1  
公共数据..... D500 和 D501  
环路 1 用数据..... D502 至 D511  
环路 2 用数据..... D512 至 D521
- (5) 将 I/O 数据设置到下列软元件中。\*2  
公共数据..... D600 至 D609  
环路 1 用数据..... D610 至 D627  
环路 2 用数据..... D628 至 D645
- (6) 在顺控程序中将环路 1 和环路 2 的 SV 值设置为以下的值  
环路 1 ..... 600  
环路 2 ..... 1000
- (7) 将下列软元件用作 PID 控制开始/停止指令及对 AD57 的监视指令。  
PID 控制开始指令 ..... X0  
PID 控制停止指令 ..... X1  
对 AD57 的监视指令..... X2
- (8) 在 0 至 2000 的范围之内设置 A68AD 和 A62DA 的数字值。

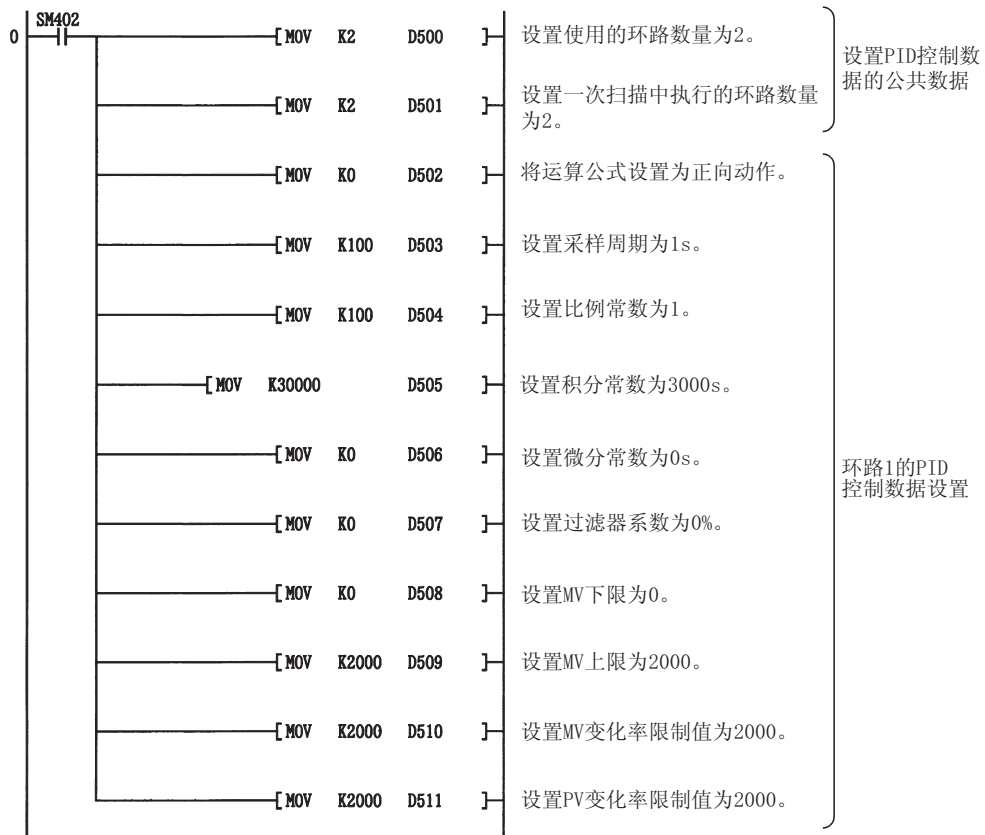
备注
----

\*1: 有关 PID 控制用数据的详细内容，请参阅 5.1 节。

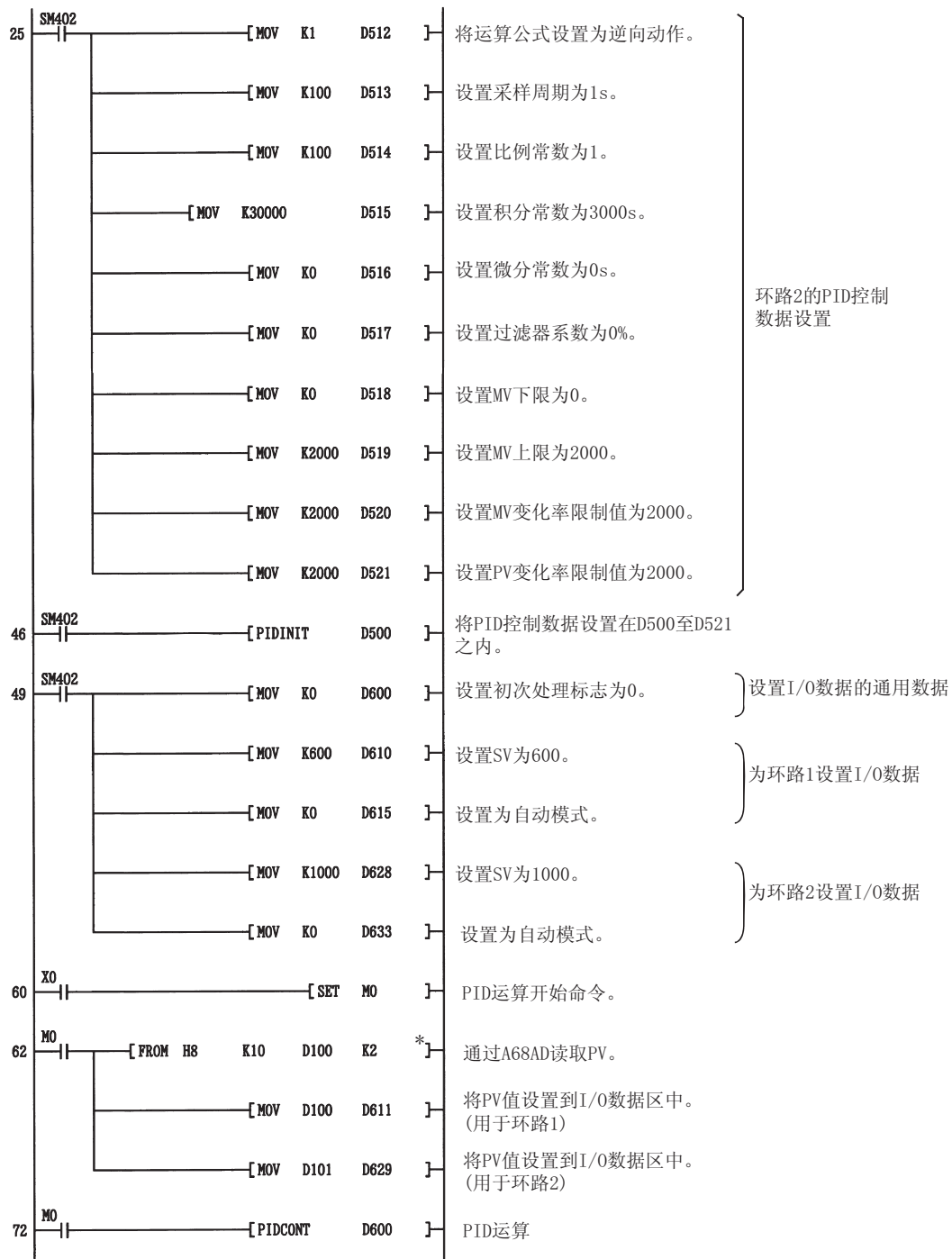
\*2: 有关 I/O 数据的详细内容，请参阅 5.2 节。

## 9. 完全微分 PID 控制指令和程序示例

### [程序示例]

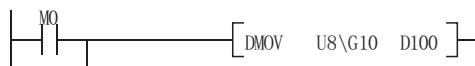


## 9. 完全微分 PID 控制指令和程序示例

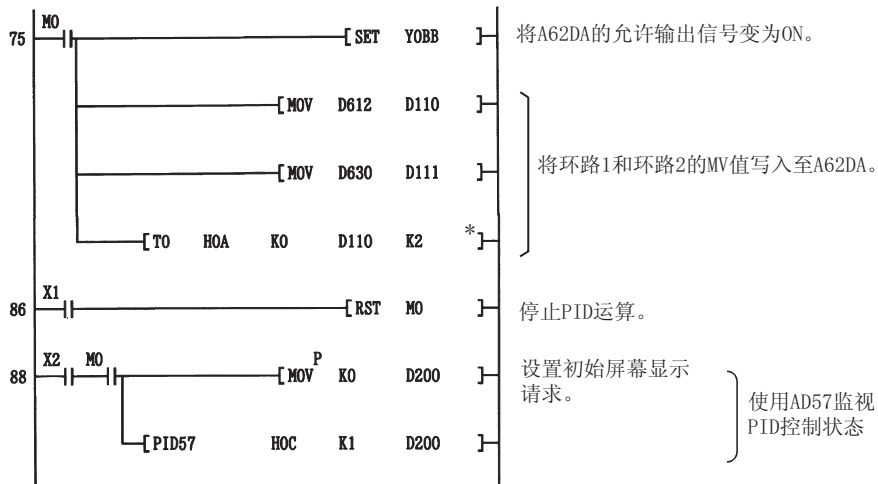


### 备注

\*: 也可以使用特殊功能模块软元件创建程序。  
 在这种情况下下的梯形图中的格式如下所示。



## 9. 完全微分 PID 控制指令和程序示例



### 备注

\*: 也可以使用特殊功能模块软元件创建程序。  
 在情况下的梯形图中的格式如下所示。

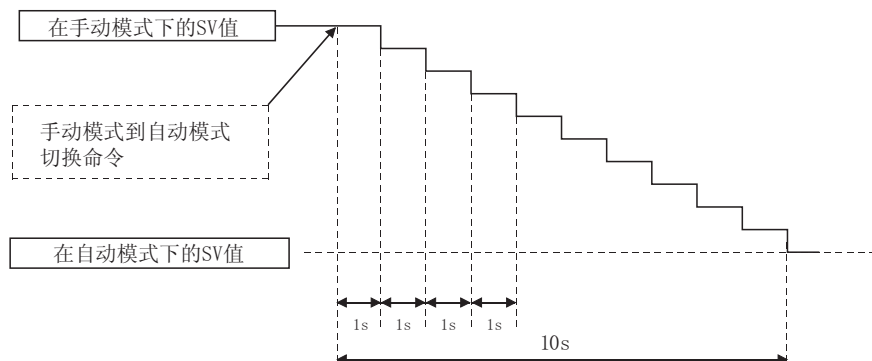


9.3.3 在自动和手动间切换 PID 控制模式的程序示例

下面介绍执行 PID 运算时在自动模式和手动模式间切换的程序示例。

[程序条件]

- (1) 有关系统配置的详细内容，请参阅 9.3.1 项。
- (2) 执行 PID 运算的环路数为 1。
- (3) 采样周期为 1s。
- (4) 在下列软元件中设置 PID 控制用数据。  
 公共数据..... D500 和 D501  
 环路 1 用数据..... D502 至 D511
- (5) 在下列软元件中设置 I/O 数据。  
 公共数据..... D600 至 D609  
 环路 1 用数据..... D610 至 D627
- (6) 用外部数字开关设置 SV 和手动模式时的 MV。  
 SV..... X30 至 X3F  
 MV (手动控制模式)..... X20 至 X2F
- (7) 用下列软元件作为开始/停止 PID 控制指令及自动/手动切换指令。  
 PID 控制开始指令..... X0  
 PID 控制停止指令..... X1  
 对 AD57 的监视指令..... X2  
 SV 设置指令..... X3  
 手动模式下的 MV 设置指令..... X4  
 自动/手动模式切换指令..... X6  
 (OF: 自动模式; ON: 手动模式)
- (8) 在 0 至 2000 的范围之内设置 A68AD 和 A62DA 的数字值。
- (9) 将 PID 无冲击处理标志 SM774 设置成 OFF。当把控制模式从自动模式切换到手动模式时，SV 被自动地改写成 PV。因此，在将控制模式从手动重新切换到自动之前应将 SV 改写为以前自动模式控制时使用的 SV 值。但是，应按下图所示对 SV 值分 10 个阶段逐步进行改写。



使用如下所示的运算方法改写 SV。

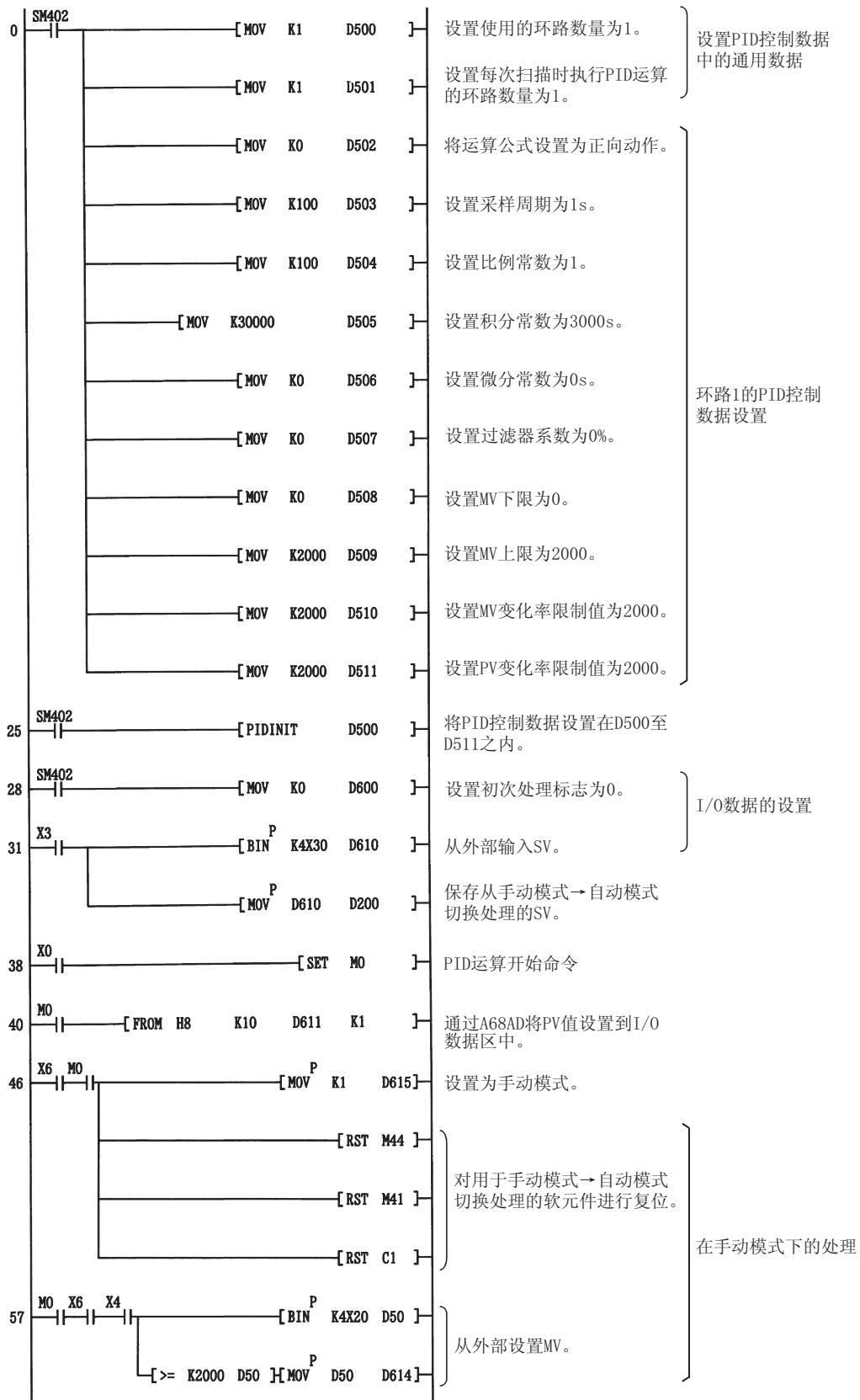
$$\frac{\left( \text{在手动模式下的SV值} \right) - \left( \text{在自动模式下的SV值} \right)}{10} = \boxed{\text{减数值}} \dots \dots \boxed{\text{余数}}$$

进行 SV 值的减法运算时，以上式的减数值逐秒进行减法运算。在进行第一次减法运算的同时减去余数。

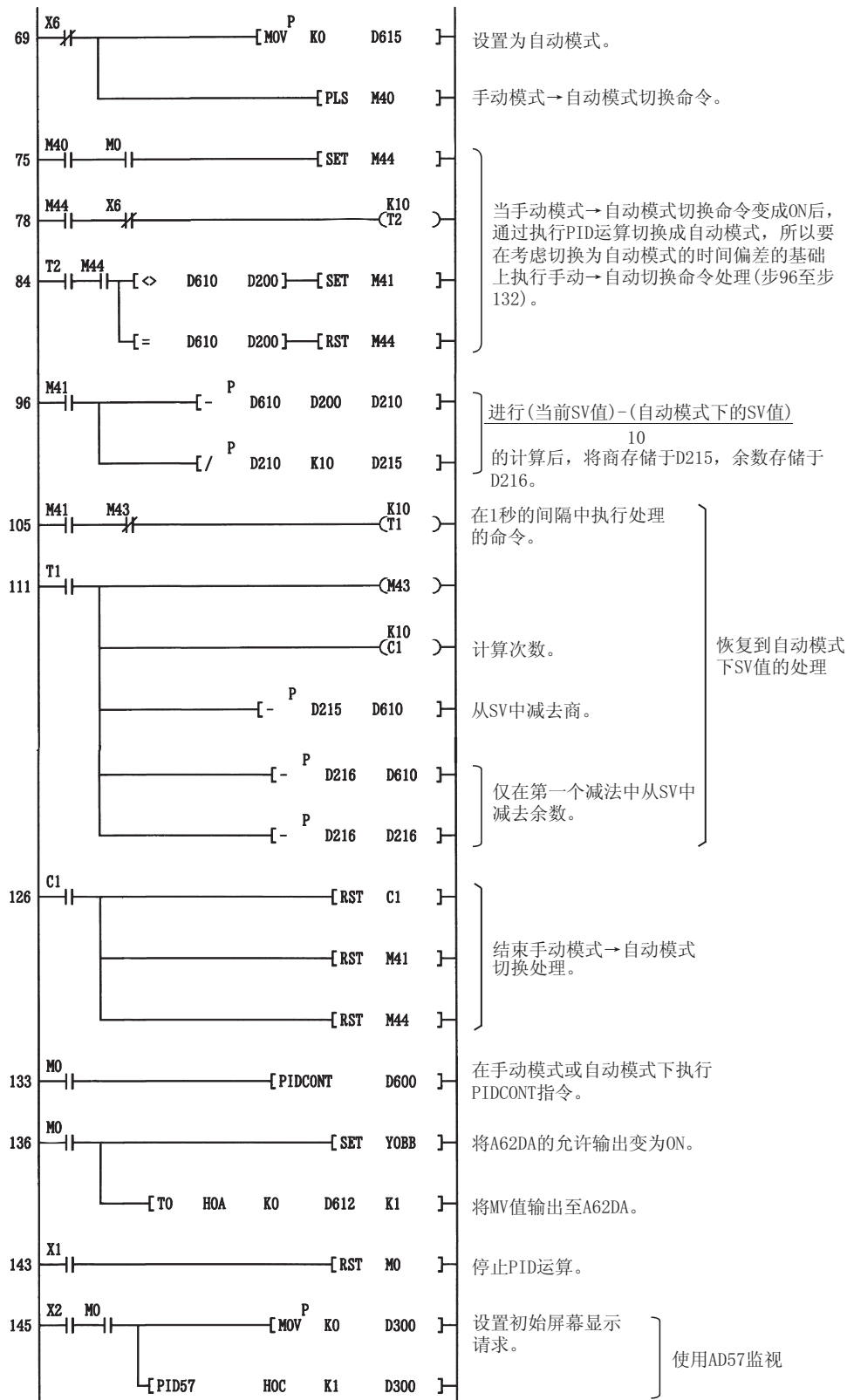


## 9. 完全微分 PID 控制指令和程序示例

### [程序示例]



[程序示例]



附录

附录 1 处理时间列表

(1) 下表说明了不完全微分 PID 控制指令的处理时间。

(a) 基本型 QCPU、高性能型 QCPU、冗余 CPU、通用型 QCPU

指令名称	条件		处理时间 (μs)					
			Q00JCPU	Q00CPU	Q01CPU	Q02CPU	QnHCPU	QnPRHCPU
S. PIDINIT	1 个环路		115.0	97.0	88.5	64.5	28.0	28.0
	8 个环路		250.0	210.0	190.0	-	-	-
	32 个环路		-	-	-	410.0	180.0	180.0
S. PIDCONT	1 个环路	第一次	395.0	335.0	300.0	215.0	92.0	92.0
		其它	350.0	300.0	270.0	190.0	81.5	81.5
	8 个环路	第一次	2250.0	1850.0	1700.0	-	-	-
		其它	1950.0	1650.0	1500.0	-	-	-
	32 个环路	第一次	-	-	-	4550.0	1950.0	1950.0
		其它	-	-	-	4450.0	1850.0	1850.0
S. PIDSTOP	1 个环路		79.5	66.0	61.0	25.0	11.0	11.0
S. PIDRUN	1 个环路		120.0	99.5	89.5	60.0	26.0	26.0

指令名称	条件		处理时间 (μs)							
			Q00JCPU Q00UCPU Q01UCPU		Q02UCPU		Q03UDCPU Q03UDECPU		Q04UDHCPU、Q06UDHCPU、 Q10UDHCPU、Q13UDHCPU、 Q20UDHCPU、Q26UDHCPU、 Q04UDEHCPU、Q06UDEHCPU、 Q10UDEHCPU、Q13UDEHCPU、 Q20UDEHCPU、Q26UDEHCPU、 Q50UDEHCPU、Q100UDEHCPU	
			最小值	最大值	最小值	最大值	最小值	最大值	最小值	最小值
S. PIDINIT	1 个环路		17.5	39.9	14.2	48.2	14.2	22	11.4	18.8
	8 个环路		-	-	-	-	-	-	-	-
	32 个环路		295	376	230.1	298	231	261	164	183
S. PIDCONT	1 个环路	第一次	123		114	131	50.5	52.5	46.8	47.5
		其它	90.4		77.4	111	39.5	47.2	35.8	42.1
	8 个环路	第一次	-		-	-	-	-	-	-
		其它	-		-	-	-	-	-	-
	32 个环路	第一次	1121		1082	1115	1036	1041	927	931
		其它	1046		852	883	820	842	728	744
S. PIDSTOP	1 个环路		7.3	14.2	5.6	17.4	5.5	7.5	4.9	6.9
S. PIDRUN	1 个环路		5.5	11.0	4.9	10.6	4.8	6.1	4.3	5.6
S. PIDPRMW	1 个环路		18.7	62.4	13.3	33	13.0	16.7	10.7	14.5

(b) LCPU

指令名	条件		处理时间 ( $\mu\text{s}$ )			
			L02CPU		L26CPU-BT	
			最小值	最大值	最小值	最大值
S. PIDINIT	1个环路		14.0	27.3	11.4	18.8
	8个环路		--	--	--	--
	32个环路		244.3	277	164	183
S. PIDCONT	1个环路	第一次	64.1	66.0	46.8	47.5
		其它	53.0	55.5	35.8	42.1
	8个环路	第一次	--	--	--	--
		其它	--	--	--	--
	32个环路	第一次	1064.2	1069.5	927	931
		其它	863.2	889.8	728	744
S. PIDSTOP	1个环路		6.9	10.6	4.9	6.9
S. PIDRUN	1个环路		6.2	15.2	4.3	5.6
S. PIDPRMW	1个环路		14.7	28.9	10.7	14.5

(2) 下表说明了完全微分PID控制指令的处理时间。

(a) 基本型 QCPU、高性能型 QCPU、冗余 CPU、通用型 QCPU

指令名称	条件	处理时间 (μs)									
		Q2ASCPU Q2ACPU (S1)	Q3ACPU	Q2ASHCPU (S1) Q4ACPU Q4ARCPU	Q00J CPU	Q00 CPU	Q01 CPU	Q02 CPU	QnH CPU	QnPRHCP U	
PIDINIT	1 个环路	61	46	23	66.0	56.0	50.5	26.0	11.2	11.2	
	8 个环路	-	-	-	170.0	145.0	130.0	-	-	-	
	32 个环路	407	306	153	-	-	-	174.0	74.9	74.9	
PIDCONT	1 个环路	第一次	211	159	80	325.0	275.0	245.0	86.6	37.3	37.3
		其它	181	136	68	285.0	250.0	225.0	74.3	32.0	32.0
	8 个环路	第一次	-	-	-	2000.0	1700.0	1500.0	-	-	-
		其它	-	-	-	1700.0	1450.0	1300.0	-	-	-
	32 个环路	第一次	5086	3824	1912	-	-	-	2102.5	904.9	904.9
		其它	4894	3680	1840	-	-	-	2036.9	876.7	876.7
PID57	1 个环路	第一次	9629	7240	3620	-	-	-	-	-	-
		其它	606	456	228						
	8 个环路	第一次	9669	7270	3635						
		其它	3719	2796	1398						
PIDSTOP PIDRUN	1 个环路	11.2	8.4	4.2	22.0	18.5	17.0	4.5	1.9	1.9	
PIDPRMW	1 个环路	36	26	13	53.0	45.0	41.0	14.6	6.3	6.3	

指令名称	条件	处理时间 (μs)							
		Q00UJCPU Q00UCPU Q01UCPU		Q02UCPU		Q03UDCPU Q03UDECPU		Q04UDHCPU、Q06UDHCPU、 Q10UDHCPU、Q13UDHCPU、 Q20UDHCPU、Q26UDHCPU、 Q04UDEHCPU、Q06UDEHCPU、 Q10UDEHCPU、Q13UDEHCPU、 Q20UDEHCPU、Q26UDEHCPU、 Q50UDEHCPU、Q100UDEHCPU	
		最小值	最大值	最小值	最大值	最小值	最大值	最小值	最大值
PIDINIT	1 个环路	10.3	60.7	8.2	27.7	8.2	12.4	6.3	10.9
	8 个环路 s	-	-	-	-	-	-	-	-
	32 个环路 s	162	227	129	159	129	133	98.7	122.6
PIDCONT	1 个环路	第一次	111	101	102	40.6	41.5	36.8	
		其它	95.5	51.5	74.4	33.6	38.5	35.8	
	8 个环路	第一次	-	-	-	-	-	-	
		其它	-	-	-	-	-	-	
	32 个环路 s	第一次	909	931	939	862	872	776	785
		其它	914	764	788	736	757	622	645
PIDSTOP	1 个环路	3.3	33.4	1.8	6.7	1.8	3.4	1.3	2.9
PIDRUN	1 个环路	1	4	1.7	6.7	1.7	3.2	1.5	2.7
PIDPRMW	1 个环路	9.4	23	6.9	17.6	6.9	10.3	5.9	8.9

(b) LCPU

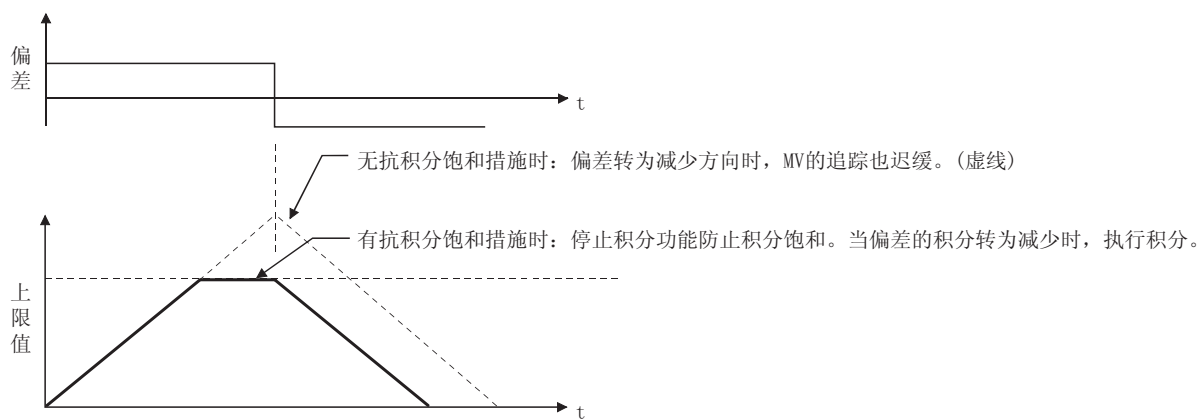
指令名	条件		处理时间 ( $\mu s$ )			
			L02CPU		L26CPU-BT	
			最小值	最大值	最小值	最大值
PIDINIT	1个环路		8.6	14.6	6.3	10.9
	8个环路		--	--	--	--
	32个环路		131.2	136.7	98.7	122.6
PIDCONT	1个环路	第一次	51.5	51.8	36.8	
		其它	50.0	50.4	35.8	
	8个环路	第一次	--		--	
		其它	--		--	
	32个环路	第一次	869.5	886.9	776	785
		其它	746.2	750.3	622	645
PIDSTOP	1个环路		2.2	4.0	1.3	2.9
PIDRUN	1个环路		2.2	4.2	1.5	2.7
PIDPRMW	1个环路		8.0	12.0	5.9	8.9

## 附录 2 抗积分饱和措施

积分饱和是指积分因子超出了饱和限制，偏差被持续累加的问题（亦称为积分器饱和）。

当发生积分饱和时，必须停止积分动作以便能够立即响应偏差的反向变化。

在 QCPU/LCPU/QnACPU 的 PID 运算指令 (PIDCONT 指令、S.PIDCONT 指令) 中，由于采取了抗积分饱和措施，因此无需停止积分动作。







# 质保

使用之前请确认以下产品质保的详细说明。

## 1. 免费质保期限和免费质保范围

在免费质保期内使用本产品时如果出现任何属于三菱责任的故障或缺陷(以下称“故障”),则经销商或三菱服务公司将负责免费维修。

但是如果需要在国内现场或海外维修时,则要收取派遣工程师的费用。对于涉及到更换故障模块后的任何再试运转、维护或现场测试,三菱将不负任何责任。

### [免费质保期限]

免费质保期限为自购买日或交货的一年内。

注意产品从三菱生产并出货之后,最长分销时间为6个月,生产后最长的免费质保期为18个月。维修零部件的免费质保期不得超过修理前的免费质保期。

### [免费质保范围]

(1) 一次故障诊断原则上由贵公司实施。

但是,根据贵公司要求三菱或三菱服务网可以进行有偿服务。在这种情况下,故障原因属于三菱方面的情况下将无偿服务。

(2) 范围局限于按照使用手册、用户手册及产品上的警示标签规定的使用状态、使用方法和使用环境正常使用的情况下。

(3) 以下情况下,即使在免费质保期内,也要收取维修费用。

- 1) 因不适当存储或搬运、用户过失或疏忽而引起的故障。因用户的硬件或软件设计而导致的故障。
- 2) 因用户未经批准对产品进行改造而导致的故障等。
- 3) 对于装有三菱产品的用户设备,如果根据现有的法定安全措施或工业标准要求配备必需的功能或结构后本可以避免的故障。
- 4) 如果正确维护或更换了使用手册中指定的消耗部件后本可以避免的故障。
- 5) 消耗部件(电池、继电器、保险丝等)的更换。
- 6) 因火灾或异常电压等外部因素以及因地震、雷电、大风和水灾等不可抗力而导致的故障。
- 7) 根据从三菱出货时的科技标准还无法预知的原因而导致的故障。
- 8) 任何非三菱或用户责任而导致的故障。

## 2. 产品停产后的有偿维修期限

(1) 三菱在本产品停产后的7年内受理该产品的有偿维修。

停产的消息将以三菱技术公告等方式予以通告。

(2) 产品停产,将不再提供产品(包括维修零件)。

## 3. 海外服务

在海外,维修由三菱在当地的海外FA中心受理。注意各个FA中心的维修条件可能会不同。

## 4. 意外损失和间接损失不在质保责任范围内

无论是否在免费质保期内,对于任何非三菱责任的原因而导致的损失、机会损失、因三菱产品故障而引起的用户利润损失、无论能否预测的特殊损失和间接损失、事故赔偿、除三菱以外产品的损失赔偿、用户更换设备、现场机械设备的再调试、运行测试及其它作业等,三菱将不承担责任。

## 5. 产品规格的改变

目录、手册或技术文档中的规格如有改变,恕不另行通知。



SH (NA) -080240C-D (1104) MEACH

MODEL: QNA/QCPU-P (PI) -C

 **三菱电机自动化(中国)有限公司**

地址：上海市虹桥路1386号三菱电机自动化中心

邮编：200336

电话：021-23223030 传真：021-23223000

网址：<http://cn.MitsubishiElectric.com/fa/zh/>

技术支持热线 **400-821-3030**



扫描二维码,关注官方微博



扫描二维码,关注官方微信

内容如有更改 恕不另行通知