User's

LG Programmable Logic Controller

GLOFA G3F – PIDA G4F – PIDA

LG Industrial Systems

◎ CONTENTS ◎

Chapter 1. INTRODUCTION

Chapter 2. SPECIFICATIONS

Chapter 3. INSTALLATION

Chapter 4. TROUBLESHOOTING

Chapter 5. DIMENSIONS

Chapter 6. FUNCTION BLOCK

Chapter 7. PROGRAMMING

SAFETY PRECAUTIONS

Be sure to read carefully the safety precautions given in data sheet and user's manual before operating the module and follow them.

The precautions explained here only apply to the G3F-PIDA and G4F-PIDA.

For safety precautions on the PLC system, see the GLOFA GM3/4 User's Manuals.

A precaution is given with a hazard alert triangular symbol to call your attention, and precautions are represented as follows according to the degree of hazard.

⚠ **WARNING** ⇨	If not provided with proper prevention, it can cause death or fatal injury or considerable loss of property.
⚠ **CAUTION** ⇨	If not properly observed, it can cause a hazard situation to result in severe or slight injury or a loss of property.

However, a precaution followed with ⚠ **CAUTION** can also result in serious conditions.

Both of two symbols indicate that an important content is mentioned, therefore, be sure to observe it.

Keep this manual handy for your quick reference in necessary.

Installation Precautions

CAUTION

- Operate the PLC in the environment conditions given in the general specifications.
- If operated in other environment not specified in the general specifications, it can cause an electric shock, a fire, malfunction or damage or degradation of the module
- Make sure the module fixing projections is inserted into the module fixing hole and fixed.
- Improper installation of the module can cause malfunction, disorder or falling.

Test Run and Maintenance Precautions

CAUTION

- Do not separate the module from the printed circuit board(PCB), or do not remodel the module.
  They can cause disorder, malfunction, damage of the module or a fire.
  When mounting or dismounting the module, perform them after the power has been turned off.
- Do not perform works while the power is applied, which can cause disorder or malfunction.

Waste Disposal Precautions

CAUTION

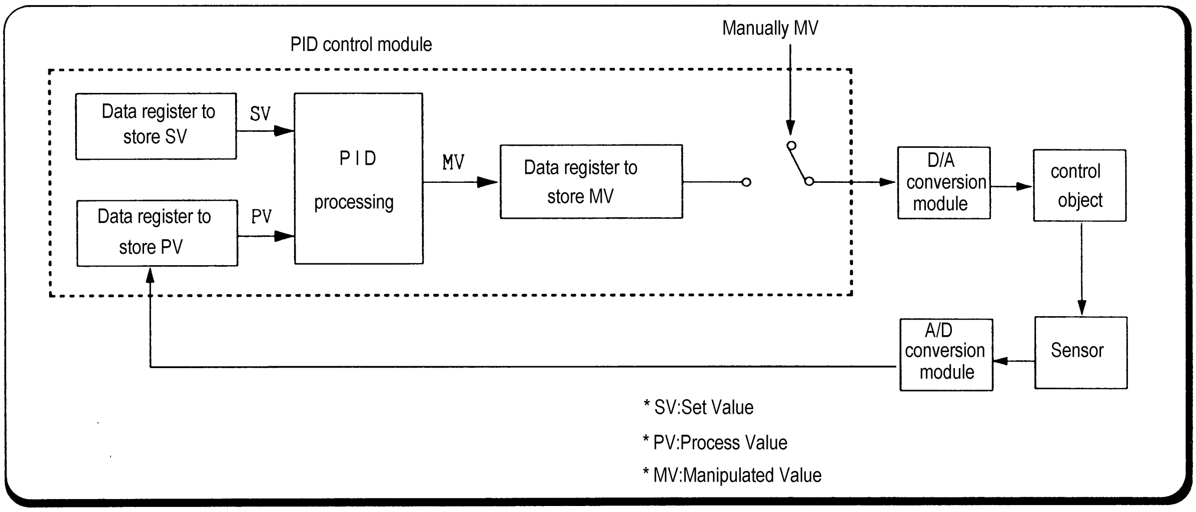
- When disposing the module, do it as an industrial waste.

Chapter 1.   INTRODUCTION

These two modules are called G3F-PIDA and G4F-PIDA. The G3F-PIDA is used with the CPU of GLOFA PLC GM1.2.3 series, and the G4F-PIDA is used with the CPU of GM4 series. Hereafter, the two modules will be commonly called the PID control module.

PID control means a control action that in order to keep the object at a value set beforehand (SV), it compares the SV with a sensor-measured value (PV) and when a difference between them is detected the controller makes PV come to be SV by adjusting output to eliminate the difference. The PID control is composed of combinations of Proportional (P), Integral (I) and Derivative (D) actions.

When a difference between SV and PV occurs, proportional, integral, differential quantities are calculated upon that difference and a MV(Manipulated Value) is output.



1.1 Features

The features of the PID control module are as follows.

1) One module can control various processes separately and at the same time.
2) Forward/reverse action selection is available.
3) Manually manipulated out (forced to be output by the user), not operation processing output, is available.
4) The number of modules available on one base unit is unlimited.

Chapter 2 .　SPECIFICATIONS

2.1　General Specifications

Table 2.1 shows the general specifications of GLOFA GM series.

<table>
<tr><th>No</th><th>Items</th><th colspan="5">Specifications</th><th>Standard</th></tr>
<tr><td>1</td><td>Operating ambient temperature</td><td colspan="5">0 ~ 55℃</td><td></td></tr>
<tr><td>2</td><td>Storage ambient temperature</td><td colspan="5">-25 ~ 75℃</td><td></td></tr>
<tr><td>3</td><td>Operating ambient humidity</td><td colspan="5">5 ~ 95%RH,　non-condensing</td><td></td></tr>
<tr><td>4</td><td>Storage ambient humidity</td><td colspan="5">5 ~ 95%RH,　non-condensing</td><td></td></tr>
<tr><td rowspan="8">5</td><td rowspan="8">Vibration</td><td colspan="5">Occasional vibration</td><td rowspan="8">IEC 1131-2</td></tr>
<tr><td>Frequency</td><td>Acceleration</td><td colspan="2">Amplitude</td><td>Sweep count</td></tr>
<tr><td>10≤f∠57 Hz</td><td>-</td><td colspan="2">0.075 mm</td><td rowspan="6">10 times in each direction for X, Y, Z</td></tr>
<tr><td>57　≤f≤150 Hz</td><td>9.8㎨ {1G}</td><td colspan="2">-</td></tr>
<tr><td colspan="4">Continuous vibration</td></tr>
<tr><td>Frequency</td><td>Acceleration</td><td colspan="2">Amplitude</td></tr>
<tr><td>10≤f∠57 Hz</td><td>-</td><td colspan="2">0.035 mm</td></tr>
<tr><td>57≤f≤150 Hz</td><td>4.9㎨ {0.5G}</td><td colspan="2">-</td></tr>
<tr><td>6</td><td>Shocks</td><td colspan="5">*Maximum shock acceleration: 147 ㎨ {15G}
*Duration time :11 ms
*Pulse wave: half sine wave pulse(3 times in each of X, Y and Z directions)</td><td>IEC 1131-2</td></tr>
<tr><td rowspan="5">7</td><td rowspan="5">Noise immunity</td><td>Square wave impulse noise</td><td colspan="4">±1,500 V</td><td></td></tr>
<tr><td>Electrostatic discharge</td><td colspan="4">Voltage :4kV(contact discharge)</td><td>IEC 1131-2
IEC 801-2</td></tr>
<tr><td>Radiated electromagnetic field</td><td colspan="4">27 ~ 500 MHz, 10 V/m</td><td>IEC 1131-2
IEC 801-3</td></tr>
<tr><td rowspan="2">Fast transient burst noise</td><td>Severity Level</td><td>All power modules</td><td>Digital I/Os (Ue ≥ 24 V)</td><td>Digital I/Os (Ue < 24 V) Analog I/Os communication I/Os</td><td rowspan="2">IEC 1131-2
IEC 801-4</td></tr>
<tr><td>Voltage</td><td>2 kV</td><td>1 kV</td><td>0.25 kV</td></tr>
<tr><td>8</td><td>Operating atmosphere</td><td colspan="5">Free from corrosive gases and excessive dust</td><td></td></tr>
<tr><td>9</td><td>Altitude for use</td><td colspan="5">Up to 2,000m</td><td></td></tr>
<tr><td>10</td><td>Pollution degree</td><td colspan="5">2 or lower</td><td></td></tr>
<tr><td>11</td><td>Cooling method</td><td colspan="5">Self-cooling</td><td></td></tr>
</table>

[Table 2.1] General specifications

REMARK

1) IEC(International Electrotechnical Commission)

: The international civilian organization which produces standards for electrical and electronics industry.

2) Pollution degree

: It indicates a standard of operating ambient pollution level.

The pollution degree 2 means the condition in which normally, only non-conductive pollution occurs. Occasionally, however, a temporary conductivity caused by condensation shall be expected.

2.2 Performance Specifications

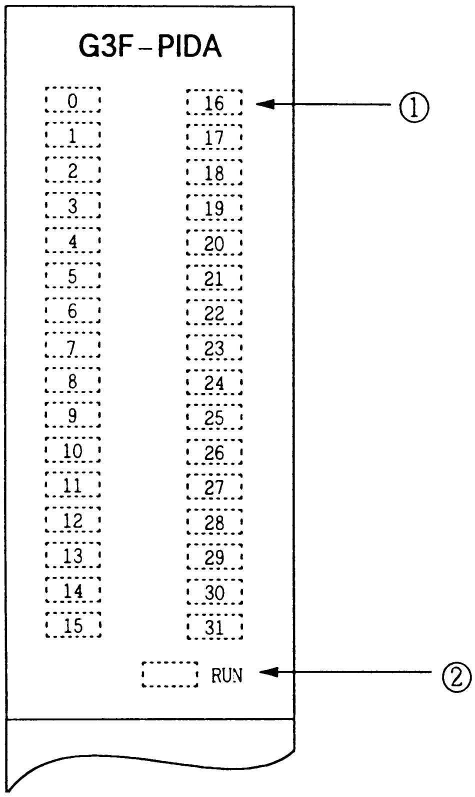
Table. 2.2 shows performance specifications of the PID control module.

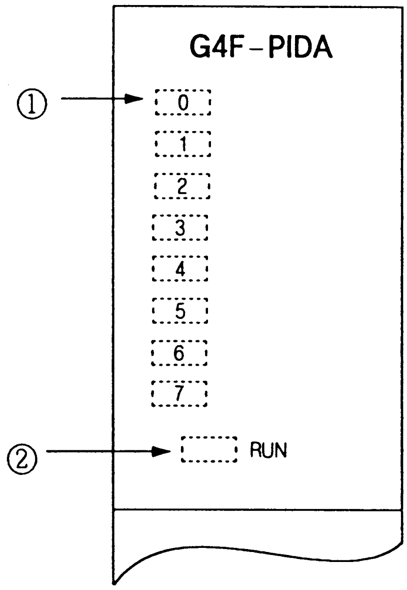
Items		Specifications	
		G3F-PIDA	G4F-PIDA
Setting range of PID constants	Proportional constant (P)	0.01 ~ 100.00 (When integral and derivative constants are set to 0.0 sec, proportional action is applied.)	
	Integral constant (I)	0.0 ~ 3000.0 sec (When integral constant is set to 0.0 sec, integral action shall not be applied.)	
	Derivative constant (D)	0.0 ~ 3000.0 sec (When derivative constant is set to 0.0 sec, derivative action shall not be applied.)	
Setting range : SV (Set Value)		0 ~ 16,000	
Input range : PV (Process Value)		0 ~ 16,000	
Output range : MV (Manipulated Value)		0 ~ 16,000	
Setting range : M_MV (Manually Manipulated Value)		0 ~ 16,000	
LED	RUN / STOP	RUN : The run LED of corresponding loops ON STOP : The run LED of corresponding loops OFF	
	NORMAL/ERROR	Normal : RUN LED ON Error : RUN LED flickering	
Number of PID control loops		32 loops	8 loops
Control action		Forward/Reverse action control is available.	
Control cycle		0.1 sec	
Processing type		Measured value derivative type (Pre-derivative type)	
Internal current consumption		0.3 A	0.2 A
Weight		370 g	190 g

[Table. 2.2 Performance Specifications]

2.3 Names of Parts and Functions

The following gives names of parts :





No.	Descriptions
①	Loop Run LED It shows the PID control module run status. ● ON : The corresponding loop is running. ● OFF : The corresponding loop is running. ● Flickering : Error status. Error Value is displayed.
②	RUN LED It shows the PID module Operating status. ● ON: Normal ● Flickering : Error

2.4 PID Control Action

2.4.1 Processing type

1) Velocity type

Velocity type is a processing that in PID processing, the present Manipulated Value(MV) is obtained by adding the calculated variation of MV (∆MV) to the previous MV

$$MV_n = MV_{n-1} + \Delta MV_n$$

MV_n : Present Manipulated Value

MV_{n-1} : Previous Manipulated Value

ΔMV_n : Variation of the Previous Manipulated Value

2) Measured Value Derivative Type (Pre-derivative)

Measured value derivative processing, in PID processing, uses the process value(PV) for the derivative term. Generally, PID processing, when a deviation occurs, operates toward the direction in which the deviation will be reduced.
The deviation occurs due to alteration of set value(SV) or outside disturbances. Therefore, if the deviation is used in the derivative processing, the output of the derivative term changes rapidly when the deviation occur due to alteration of set value (SV). So, to prevent raid changes like that, this processing uses the process value(PV) for the derivative term.

$$MV_n = MV_{n-1} + K_p \times (E_n - E_{n-1}) + K_p \times S/K_I \times E_n + K_p \times K_d/S \times (2PV_n - PV_{n-1} - PV_{n-2})$$

MV_n : Present Manipulated Value

MV_{n-1} : Previous Manipulated Value

ΔMV_n : Variation of the Previous Manipulated Value

E_n : Present Deviation

E_{n-1} : Previous Deviation

K_p : Proportional Constant

K_i : Integral Constant

K_d : Derivative Constant

S : Control Cycle (100ms)

PV_n : Present Process Quantity (Present Value)

PV_{n-1} : One-step previous Process Quantity (Present Value)

PV_{n-2} : Two-step previous Process Quantity (Present Value)

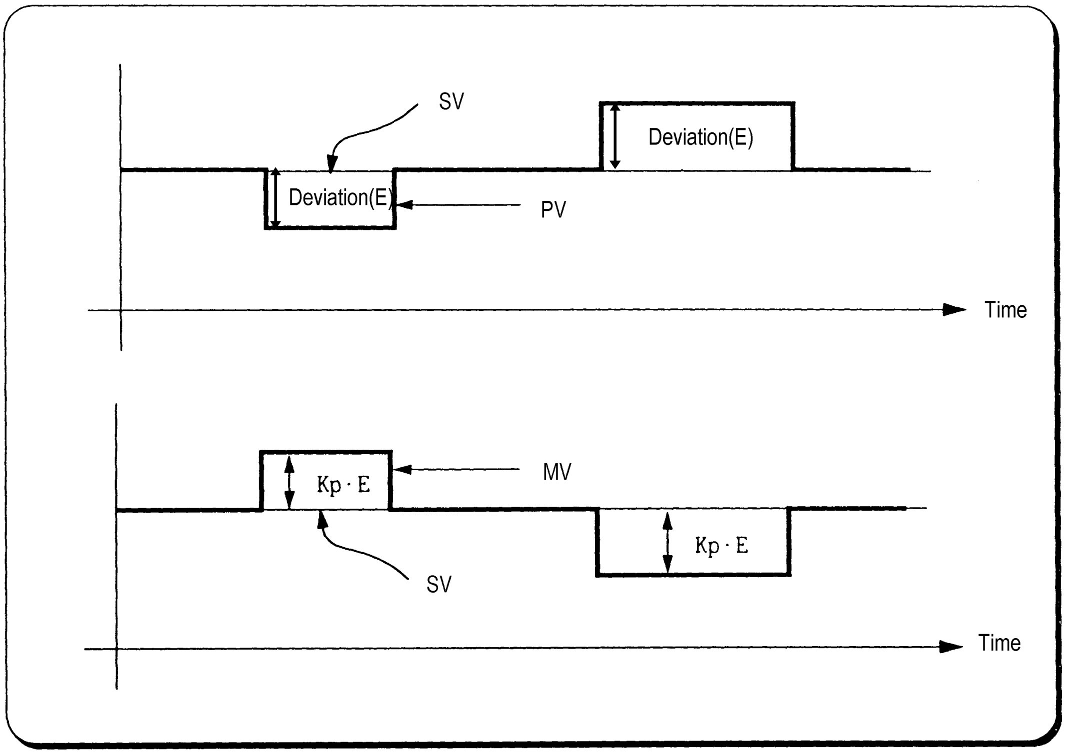
2.4.2 Control Action

1) Proportional Action (P Action)

(1) P action means a control action that obtains a MV which is proportional to the deviation (E: the difference between SV and PV).
(2) The expression which denotes the change relationship of E to MV in P action is shown as follows:

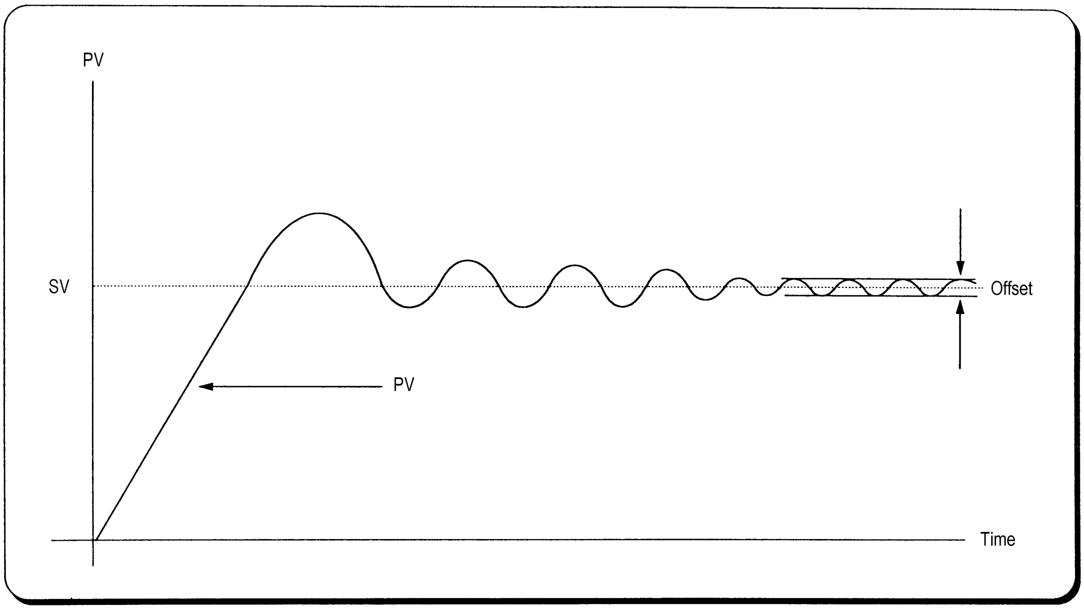
$$MV = Kp \times E$$

where Kp is a proportional constant and means gain.
(3) When deviation occurs, the MV by P action is shown in Fig. 2.1.

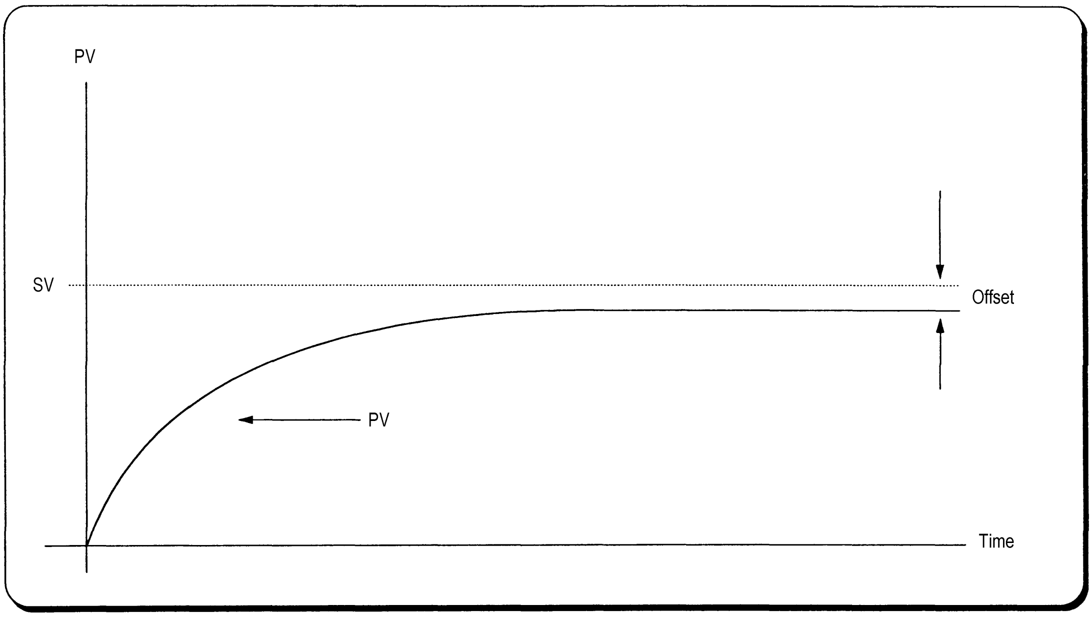


[Fig. 2.1]   MV with the proportional action

(4) As shown in Fig. 2.1, the larger the proportional constant Kp the larger the MV,   that is, the stronger the P action when the deviation(E) is same . Also, the smaller the Kp the smaller the MV after P action.
(5) If the Kp is too large, PV reaches SV swiftly but can make bad effects like oscillations shown in Fig. 2.2 and cause damage in control stability.
(6) If the Kp is too small, oscillations do not occur but the velocity with which PV reaches SV slows down and offset can happen as shown in Fig. 2.3.
(7) Manipulated Value varies within 0 to 16,000.



[Fig. 2.2] When the proportional constant Kp is large.

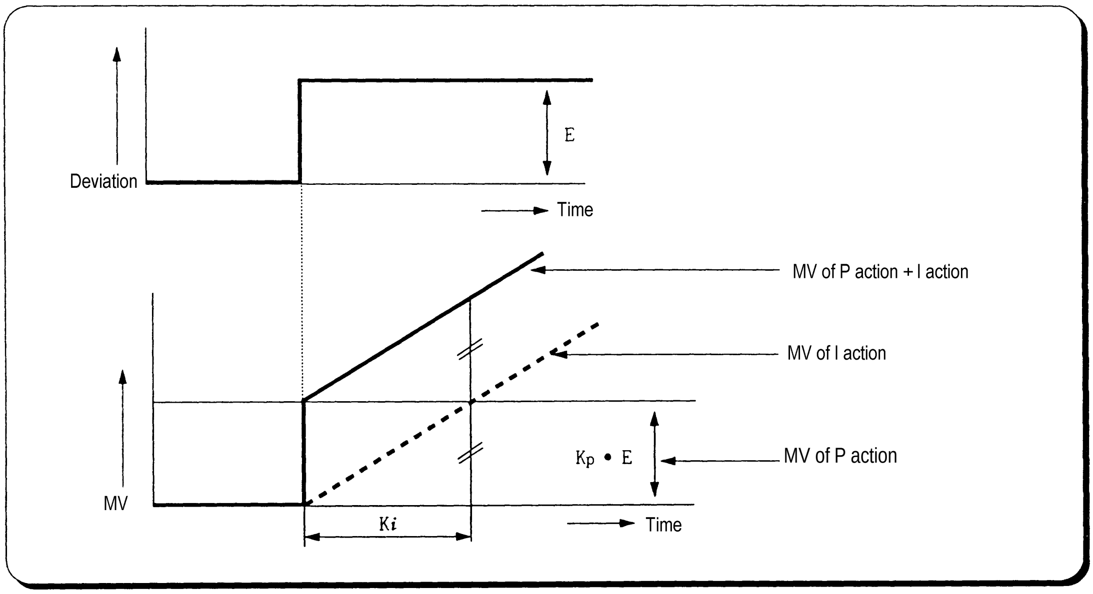


[Fig. 2.3] When the proportional constant Kp is small.

2) Integral Action (I Aaction)

(1) When a deviation(E) occurs between SV and PV, Integral action continuously adds the deviation to or subtracts it from the MV in accordance time in order to eliminate the deviation
When a deviation is small it is not expected that the MV will be changed by P action but I action will eliminate it.
Therefore, the offset which occurs in P action can be eliminated by I action.
(2) The period of the time from when the deviation has occurred in I action to when the MV of I action become that of P action is called Integration time and represented as Ki.
(3) Integral action when a given deviation has occurred is shown as the following Fig. 2.4.

MV

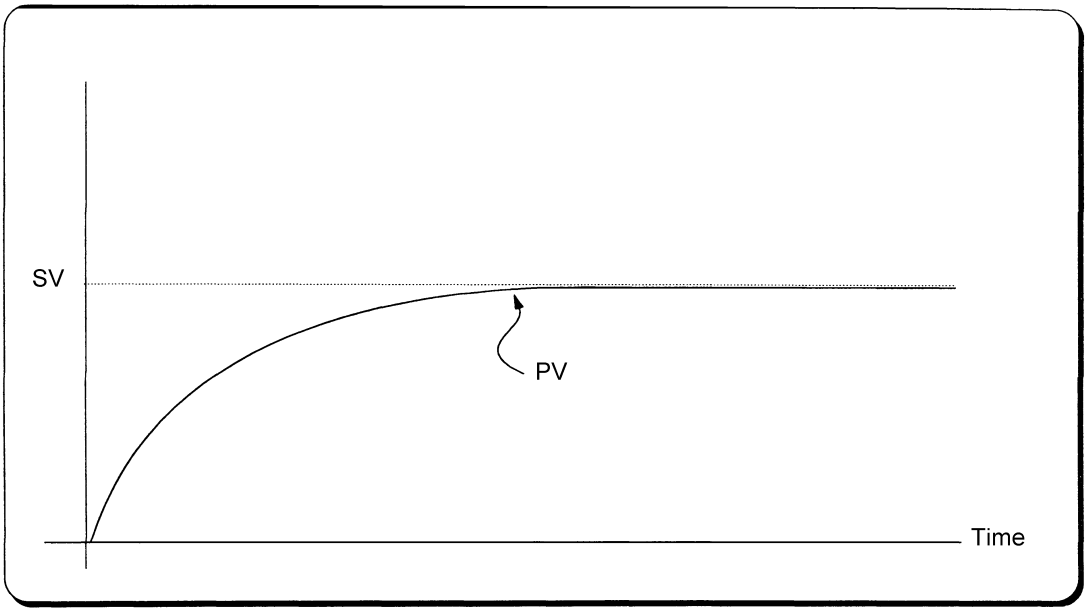


[Fig. 2.4] Integral action at a constant deviation

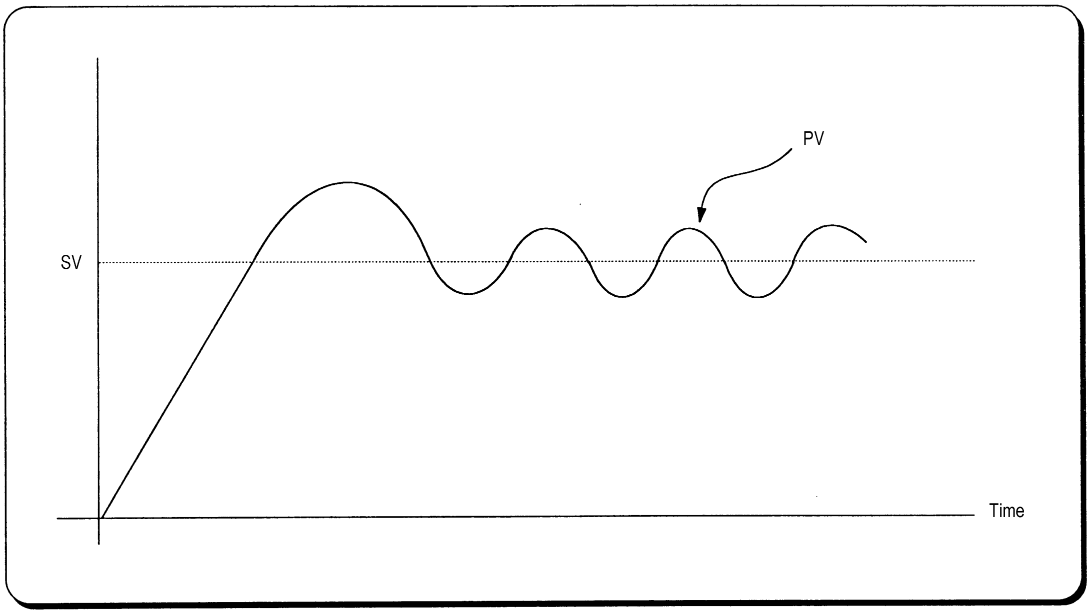
(4) Expression of Integral Action is as follows:

$$MV = P \times E + P \times \frac{1}{K_i} \times \int E\,dt$$

As shown in the expression, Integral action can be made stronger or weaker by adjusting integration time (K_i) in I action.
That is, the more the integration time (the longer the integration time) as shown in Fig. 2.5, the lesser the quantity added to or subtracted from the MV and the longer the time needed for the PV to reach the SV.
As shown in Fig. 2.6, when the integration time given is short the PV will approach the SV in short time since the quantity added or subtracted become increased. But, If the integration time is too short then oscillations occurs, therefore, the proper P.I value is requested.
(5) Integral action is used in either PI action in which P action combines with I action or PID action in which P and D actions combine with I action.



[Fig. 2.5] When a long integration time is given.



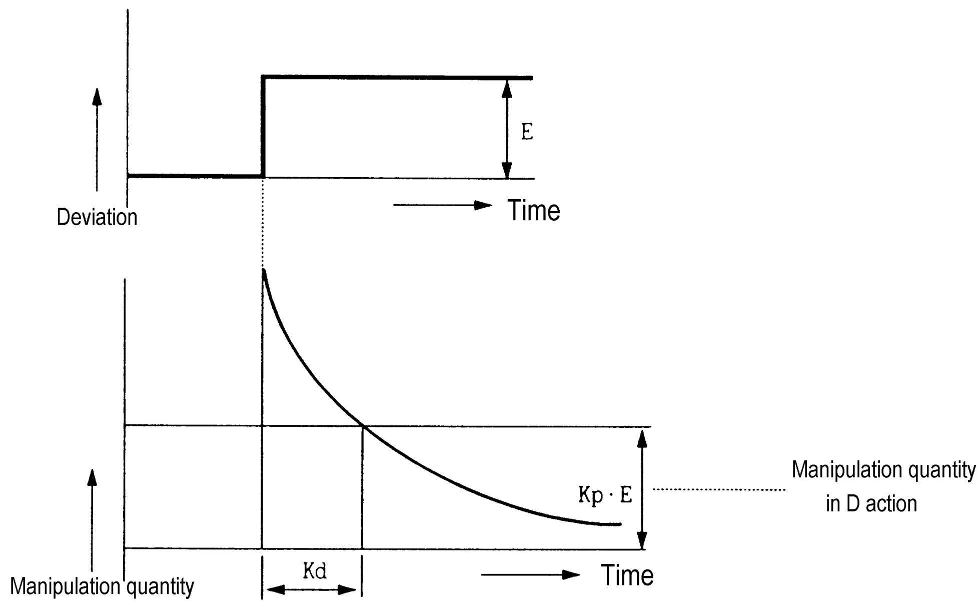
[Fig. 2.5] When a short integration time is given.

3) Derivative Action (D Action)

(1) When a deviation occurs due to alteration of SV or external disturbances, D action restrains the changes of the deviation by producing MV which is proportioned with the change velocity (a velocity whose deviation changes at every constant interval) in order to eliminate the deviation.
- D action gives quick response to control action and has an effect to reduce swiftly the deviation by applying a large control action (in the direction that the deviation will be eliminated) at the earlier time that the deviation occurs.
- D action can prevent the large changes of control object due to external conditions.

(2) The period of time from when the deviation has occurred to when the MV of D action become the MV of P action is called derivative time and represented as Kd.

(3) The D action when a given deviation occurred is shown as Fig. 2.7.



[Fig. 2.7] Derivative action at a constant deviation

(4) The expression of D action is represented as follows:

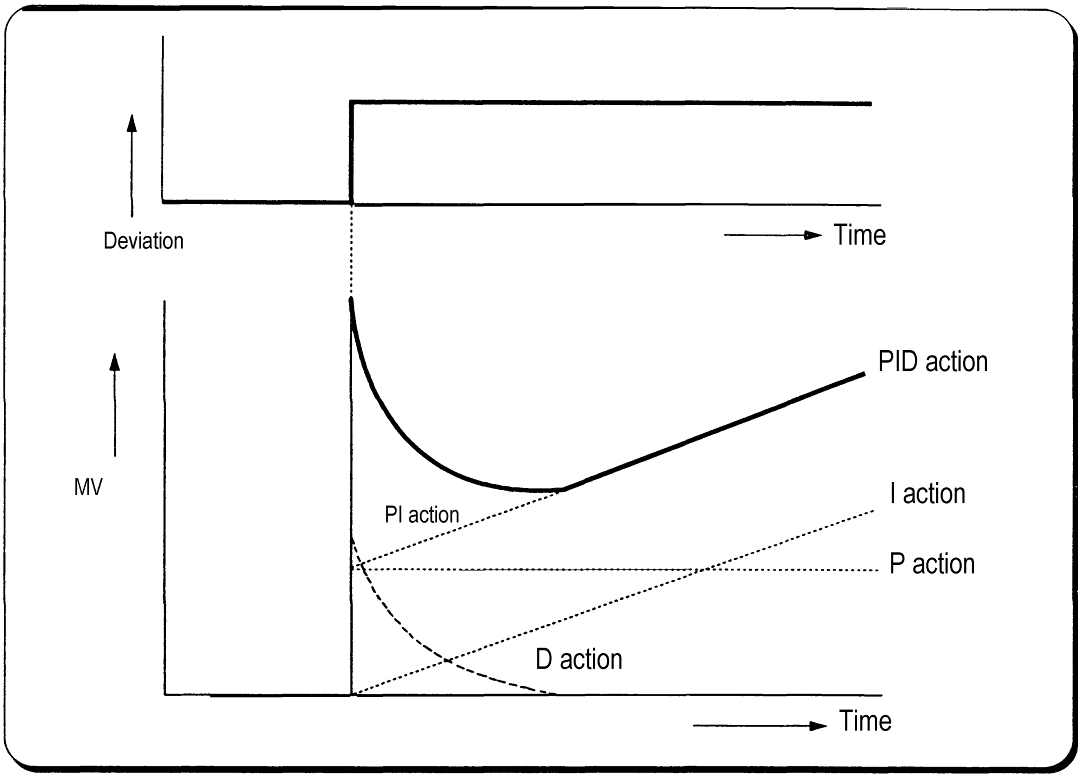
$$MV = K_p \times E + K_p \times \frac{dE}{dt}$$

- In this expression, an output proportional with the variation rate of deviation is added to P action quantity.
- If the derivative time is increased then P action is strengthened.
- D action is applied when a change of deviation occurs and the deviation at normal state become 0. D action, therefore, do not reduce offset.

(5) D action is used in either PD action in which P action combines with D action or PID action in which P and I actions combine with D action.

4) PID Action

(1) PID action controls the control object with the manipulation quantity produced by (P+I+D) action.
(2) PID action when a given deviation has occurred is shown as the following Fig. 2.8.



[Fig. 2.8] PID action at a constant deviation

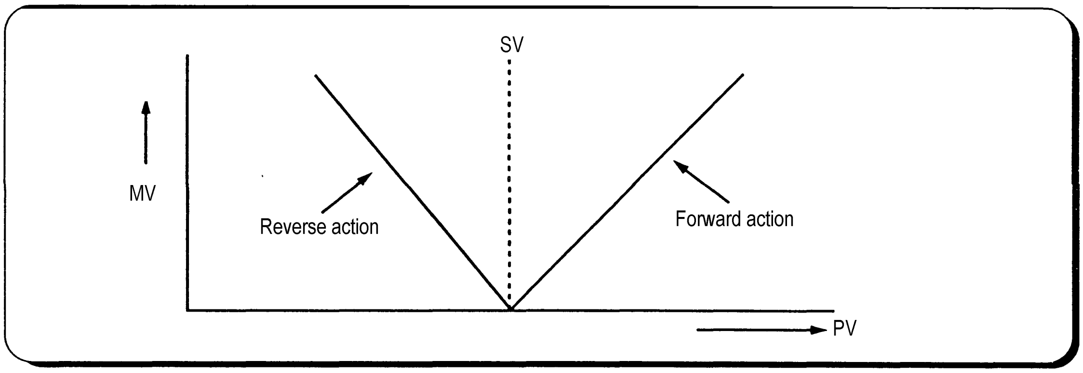
5) PID Processing Expression

PID expressions are of measured value derivative type.

Expressions	Parameters names
$E_n = SV - PV_n$ $MV_n = MV_{n-1} + K_p \times (E_n - E_{n-1})$ $+ K_p \times S/K_I \times E_n$ $+ K_p \times K_d/S \times (2PV_n - PV_{n-1} - PV_{n-2})$	**MVn** : Present Manipulated Value **MVn-1** : One-step-previous Manipulated Value **En** : Present deviation **En-1** : Previous deviation **Kp** : Proportional constant **Ki** : Integral constant **Kd** : Derivative constant **S** : Control cycle (100 ms) **PVn** : Process value **PVn-1** : One-step-previous Process Value **PVn-2** : Two-step-previous Process value

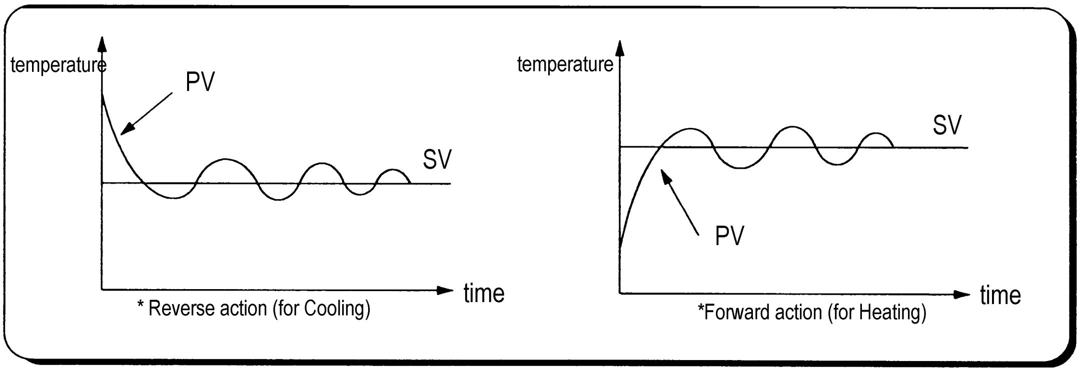
6) Forward/Reverse Actions

(1) PID control has two kinds of action, forward action and reverse action.
  a) Forward action makes PV reach SV by outputting MV when PV is less than SV.
  b) Reverse action makes PV reach SV by outputting MV when PV is more than SV.
(2) A diagram in which forward and reverse actions are drawn using MV, PV and SV is shown as Fig. 2.9.



[Fig. 2.9] Forward and reverse action with MV, PV and SV

(3) Fig 2.10 shows examples of process control by forward and reverse actions, respectively.



[Fig. 2.9] Examples of process control by forward and reverse actions

Chapter 3. INSTALLATION

3.1 Installation Ambience

This module has high reliability regardless of its installation ambience. But be sure to check the following for system in higher reliability and stability.

1) Ambience Requirements
   Avoid installing this module in locations, which are subjected or exposed to:
   - Water leakage and dust a large amount of dust, powder and other conductive power, oil mist, salt, of organic solvent exists.
   - Mechanical vibrations of impacts are transmitted directly to the module body.
   - Direct sunlight.
   - Dew condensation due to sudden temperature change.
   - High or low temperatures (outside the range of 0-55℃)

2) Installing and Wiring
   - During wiring or other work, do not allow any wire scraps to enter into the PLC
   - Install it on locations that are convenient for operation.
   - Make sure that it is not located near high voltage equipment on the same panel.
   - Make sure that the distance from the walls of duct and external equipment be 50 mm or more.
   - Be sure to be grounded to locations that have good noise immunity.

3.2 Handling Precautions

From unpacking to installing the PID control module, be sure to check the following:

1) Do not drop it off, and make sure that strong impacts should not be applied.
2) Do not dismount printed circuit boards from the case. It can cause malfunctions.
3) During wiring, be sure to check any foreign matter like wire scraps should not enter into the upper side of the PLC, and in the event that foreign matter entered into it, always eliminate it.
4) Be sure to disconnect electrical power before mounting or dismounting the module.

Chapter 4. TROUBLESHOOTING

The followings explain errors that could occur during operating the PID control module and their troubleshooting.

4.1 Errors Indicated by RUN LED Flickering

Errors indicated by PID control module RUN LED flickering are given below.

RUN LED Status	Error Type	Loop RUN LED status
Flickering (cycle: 0.1 sec)	WDT Error	Loop “0” RUN LED ON
Flickering (cycle: 0.2 sec)	System Error	All Loops RUN LED OFF
	Buffer Memory Error	Loop “1” RUN LED ON

4.2 Troubleshooting Procedure

4.2.1 RUN LED Flickering

RUN LED flickering

↓

Is the PID control module is correctly mounted on the base unit?

No → Mount correctly the PID control module.

Yes ↓

The PID control module hardware defect. If the loop run LED is turned ON, write down their error numbers and see Section 4.2.5.

4.2.2 RUN LED Off

RUN LED off

↓

Is the PID control module is correctly mounted on the base unit?

No → Mount correctly the PID control module.

Yes ↓

See Section 4.2.5.

4.2.3 Unreadable Processing Result of PID control module

Is RUN LED turned OFF?

- Yes → See Section 4.2.2.
- No ↓

Is RUN LED flickering?

- Yes → See Section 4.2.1.
- No ↓

Are the numbers of the loops used in the processing correctly specified?

- No → Specify correctly the loop numbers.
- Yes ↓

See Section 4.2.5.

4.2.4 Run LED of enabled Loops Off

Is RUN LED turned OFF?

- Yes → See Section 4.2.2.
- No ↓

Is RUN LED flickering?

- Yes → See Section 4.2.1.
- No ↓

Are the loops that should be executed correctly specified?

- No → Check the numbers of the loops which are actually being executed and specify them correctly
- Yes ↓

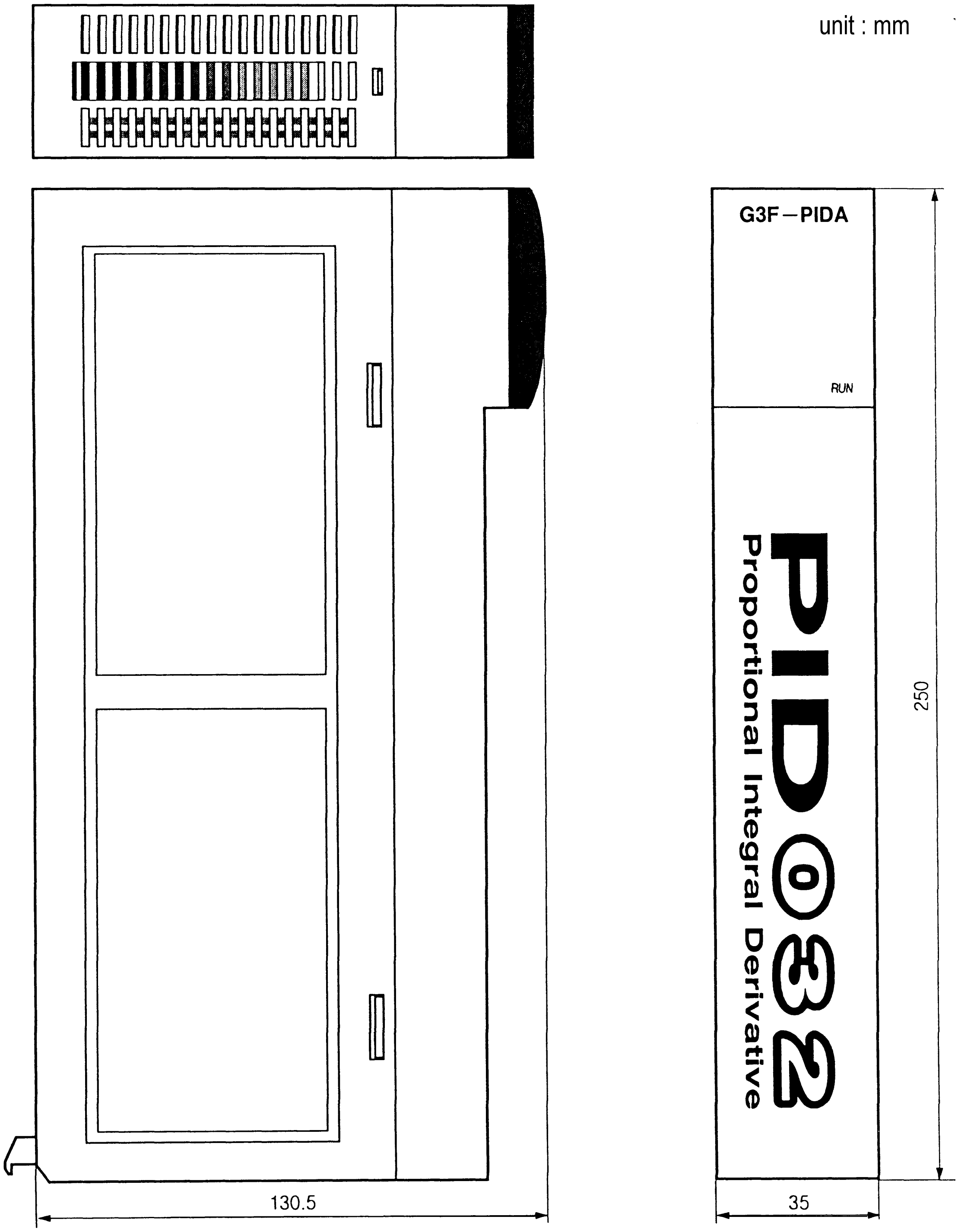
See Section 4.2.5.

4.2.5 PID Control Module Hardware Defect

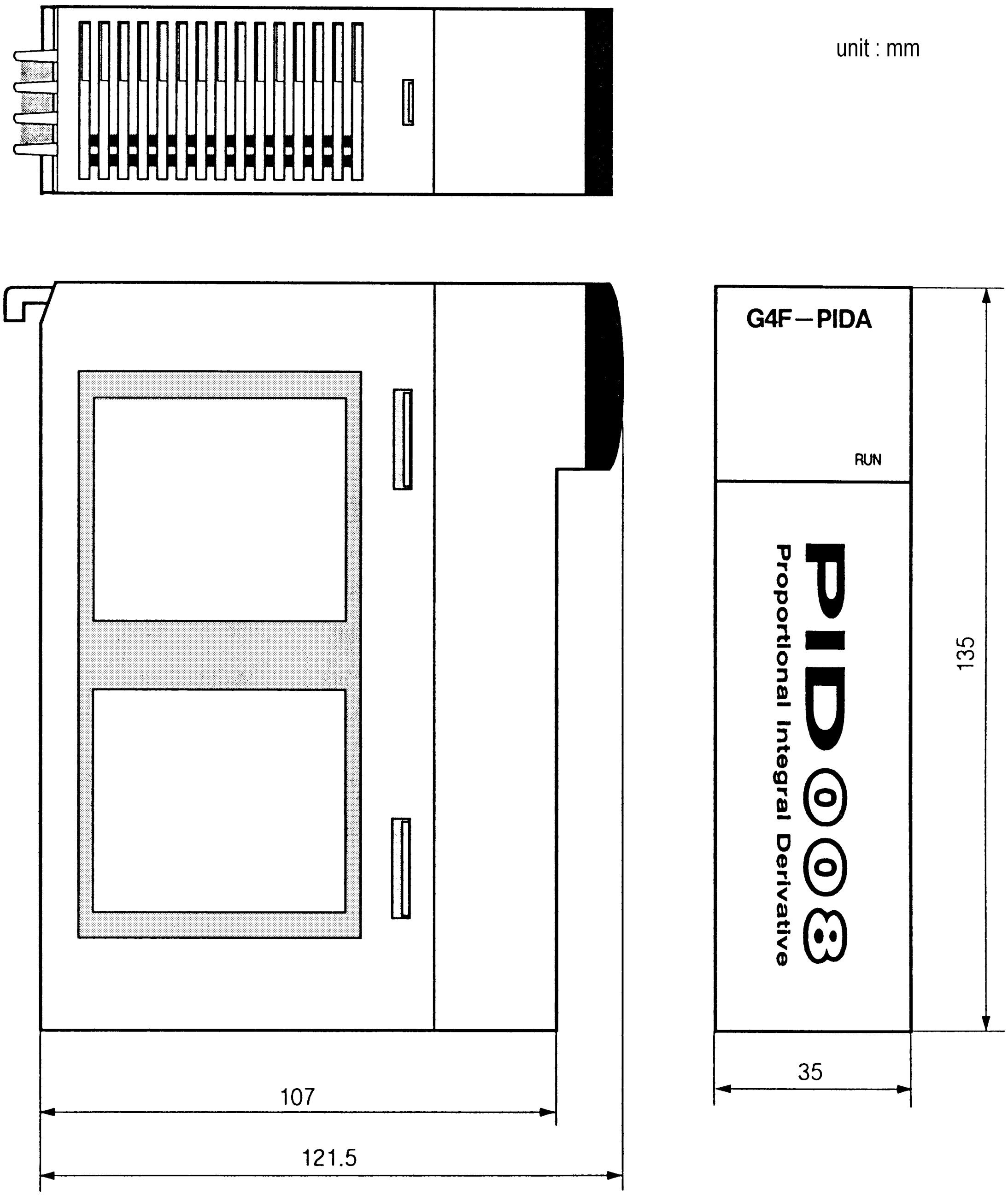
PID control module hardware defect. Contact the nearest agency or service station.

Chapter 5.   DIMENSIONS

5.1   G3F-PIDA Dimensions.



5.2 G4F-PIDA Dimensions



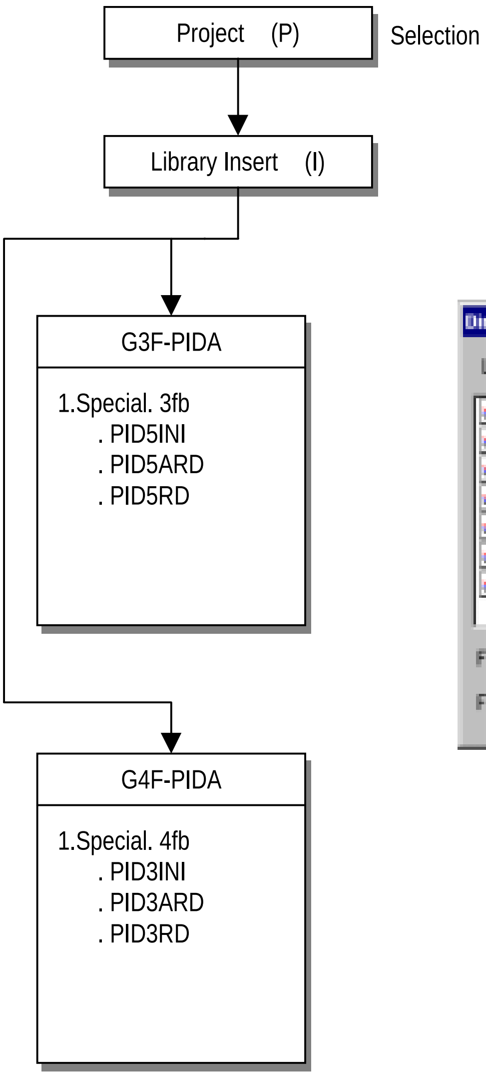
Chapter 6. FUNCTION BLOCKS

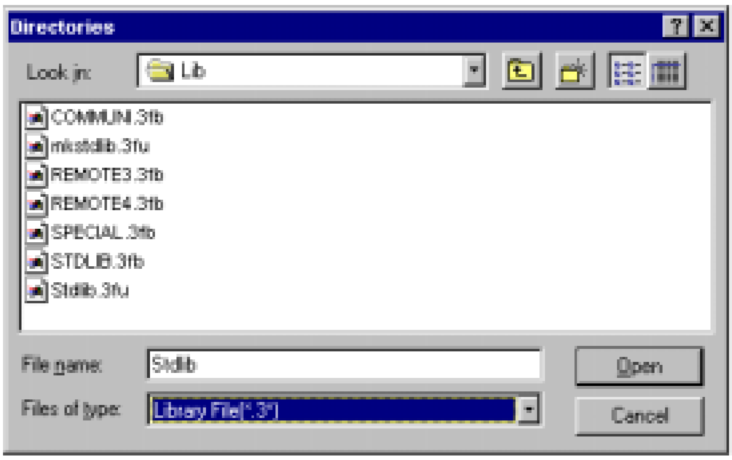
The followings explain the function blocks for the PID control module used on the GMWIN
The types of function block are given here.

No.	G3F-PIDA	G4F-PIDA	Function
1	PID5INI	PID3NI	Module Initialization
2	PID5ARD	PID3ARD	Reading the Manipulated Value (Array type)
3	PID5RD	PID3RD	Reading the Manipulated Value (Single type)

6.1 Insertion of the Function Blocks for the PID Control Module on the GMWIN.

Function blocks can be inserted with the following procedures while the GMWIN is running.
Inserting a function block is only possible when a project is open.

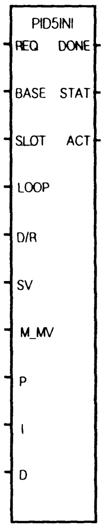
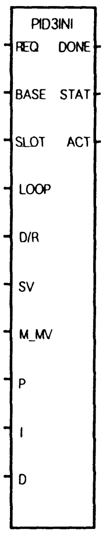




6.2 Function Blocks

6.2.1 Module Initialization (G3F-PIDA: PID5INI, G4F-PIDA:PID3INI)

Module initialization function block specifies PID control module base location, slot location, run loop enable/disable and forward/reverse action, and sets MV, M_MV and P.I.D constants for use in program.

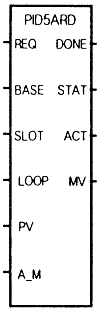
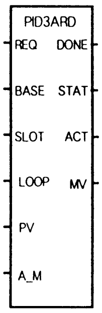
Function Block	I/O	Variable	Data Type	Descriptions
G3F－PIDA  G4F－PIDA 	In Put	REQ	BOOL	Function block execution request area - Used to request an execution of the initialization function block - If the conditions connected with this area are established while program is running and “0” changes into “1”, the initialization function block is executed
		BASE	USINT	Base location No. - Used to write the base No. where the PID control module is mounted. - Setting range: GM1 series(0~31), GM2 series(0~7), GM3/4 series(0-3)
		SLOT	USINT	Slot location No. - Used to write slot No. where the PID control module is mounted. - Setting range: 0~7
		LOOP	BOOL [Array] *Note 1	Run loop enable/disable specification - Used to enable or disable a loop for run. - Specify “1” for enabling, and “0” for disabling
		D/R	BOOL [Array] *Note 1	Forward/Reverse action specification for a run loop. - Specify “0” for forward action and “1” for reverse action.
		SV	BOOL [Array] *Note 1	Setting a SV for a run loop - Setting range: 0~16000
		M_MV	INT [Array] *Note 1	Setting a M_MV for a run loop - Setting range: 0 ~ 16000
		P	UINT [Array] *Note 1	Setting a proportional constant (0.01 ~ 100.00) for a run loop - Setting range: 0~10000 - The initialization function block not executed if the proportional constant is set to “0”, whether or not the constant is initialized in the function block.
		I	UINT [Array] *Note 1	Setting an integral constant (0.0 ~3000.0 sec) for a run loop - Setting range: 0~30000 - Integral action not executed if the integral constant is set to ‘0’.
		D	UINT [Array] *Note 1	Setting a derivative constant (0.0 ~3000.0 sec) for a run loop - Setting range: 0~30000 - Derivative action not executed if the derivative constant is set to ‘0’.
	Out Put	DONE	BOOL	Function block finished execution status - “1” is output when the initialization function block is finished with no error and “1” remains until next execution. If an error occur, ‘0’ is displayed and the operation enters into the stop state.
		STAT	USINT	Error status indication area - Used to output the number of an error when it occurs during initialization function block execution. - For description of errors, see GM Section 6.3
		ACT	BOOL [Array] *Note 1	Run loop status indication area - After the initialization function block is finished with no error, “1” is output if the loop is in normal state. But “0” is output for the disabled loops.

REMARK

* Note 1: The numbers of Array are 32 in G3F-PIDA, 8 in G4F-PIDA.

6.2.2 Manipulated Value(MV) Reading (Array type) : (G3F-PIDA:PID5ARD, G4F-PIDA:PID3ARD)

The Array type MV Reading function block execute all loops of the PID control module in a batch processing and can display the MV for run loops which is output with auto/manual run specification and a PV input.

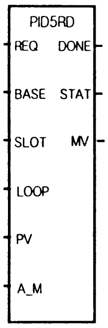
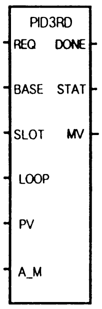
Function Block	I/O	Variable	Data Type	Descriptions
G3F－PIDA 	I	REQ	BOOL	Function block execution request area - Used to request an execution of the MV reading function block - If the conditions connected with this area are established while program is running and “0” changes into “1”, the MV reading function block is executed.
		BASE	USINT	Base location No. - Used to write the base No. where the PID control module is mounted. - Setting range: GM1 series(0~31), GM2 series(0~7), GM3/4 series(0-3)
		SLOT	USINT	Slot location No. - Used to write slot No. where the PID control module is mounted. - Setting range: 0~7
		LOOP	BOOL [Array] *Note 1	Run loop enable/disable specification - Used to enable or disable a loop for run. - Specify “1” for enabling, and “0” for disabling
		PV	INT [Array] *Note 1	Inputting a PV of the control object for a run loop - Setting range: 0~16000
		A/M	BOOL [Array] *Note 1	MV type specification for a run loop - Specify “0” for auto processing (PID processing) MV - Specify “1” for manual processing (forced processing) MV
G4F－PIDA 	O	DONE	BOOL	Function block finished execution status - “1” is output when the initialization function block is finished with no error and “1” remains until next execution. If an error occur, ‘0’ is displayed and the operation enters into the stop state.
		STAT	USINT	Error status indication area - Used to output the number of an error when it occurs during initialization function block execution. - For description of errors, see GM Section 6.3
		ACT	BOOL [Array] *Note 1	Run loop status indication area - After the initialization function block is finished with no error, “1” is output if the loop is in normal state. But “0” is output for the disabled loops.
		MV	INT [Array] *Note 1	MV data for the enabled run loops - MV output range: 0 ~ 16000

REMARK

*Note 1: The numbers of Array are 32 in G3F-PIDA, 8 in G4F-PIDA.

6.2.3 Manipulated Value(MV) Reading (Single type) : (G3F-PIDA:PID5RD, G4F-PIDA:PID3RD)

The single type MV Reading function block processes one loop of the PID control module and can display the MV for run loops which is output with auto/manual run specification and a PV input.

Function Block	I/O	Variable	Data Type	Description
G3F－PIDA 	I	REQ	BOOL	Function block execution request area - Used to request an execution of the MV reading function block - If the conditions connected with this area are established while program is running and "0" changes into "1", the MV reading function block is executed.
		BASE	USINT	Base location No. - Used to write the base No. where the PID control module is mounted. - Setting range: GM1 series(0~31), GM2 series(0~7), GM3/4 series(0-3)
		SLOT	USINT	Slot location No. - Used to write the slot No. where the PID control module is mounted. - Setting range: 0~7
		LOOP	USINT	Specifying the loop that will read MV - Setting range : G3F-PIDA: 0 to 31, G4F-PIDA: 0 to 7
		PV	INT	Inputting a PV of the control object for a run loop - Setting range: 0~16000
G4F－PIDA		A/M	BOOL	MV type specification for a run loop - Specify "0" for auto processing (PID processing) MV - Specify "1" for manual processing (forced processing) MV
	O	DONE	BOOL	Function block finished execution status - "1" is output when the initialization function block is finished with no error and "1" remains until next execution. If an error occur, '0' is displayed and the operation enters into the stop state.
		STAT	USINT	Error status indication area - Used to output the number of an error when it occurs during initialization function block execution. - For description of errors, see GM Section 6.3
		MV	INT	MV data for the enabled run loops - MV output range: 0 ~ 16000

6.3 Errors on Function Block

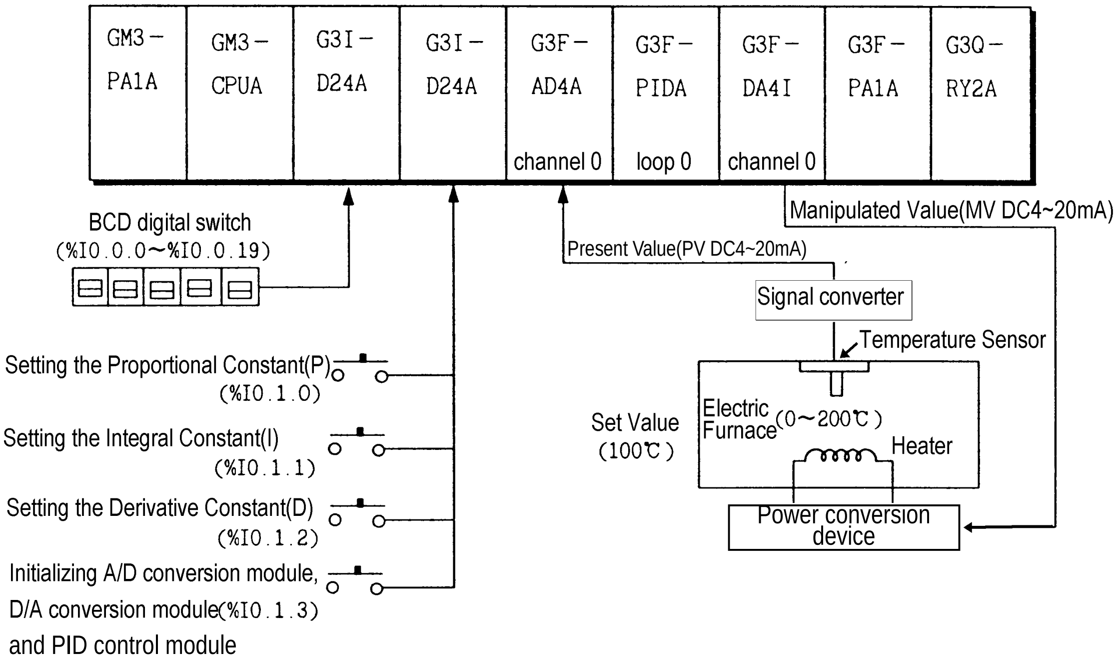
Errors indicated by an output variable STAT and their corrective actions are explained.

STAT No.	Item	Descriptions	Function Block			Corrective Action
			Initilaiza-tion	Reading		
				Array	Single	
0	Local	Normal Run status	O	O	O	—
1		Base location No. outside the setting range	O	O	O	Adjust it within the setting range (See Section GM 6.2)
2		The corresponding base unit hardware defect	O	O	O	Contact a service station
3		Slot location No. outside the setting range	O	O	O	Specify correctly the slot No. where the PID control module is mounted .
4		The specified slot has no PID control module	O	O	O	Mount the PID control module on the specified slot.
5		A module other than the PID control module is loaded on.	O	O	O	Mount the PID control module on the specified slot.
6		Loop No. outside the setting range	—	—	O	Specify correctly the No. of the run loop.
7		PID control Module hardware Defect	O	O	O	Contact a service station.
8		PID control module shared memory defect	O	O	O	Contact a service station.
9		The run loop was not specified in the Initialization function block.	—	O	O	Specify correctly run loops in the initialization function block.
10		Inputs outside the setting range	O	O	O	One or more of SV, M_MV, P, I, D and PV outside the setting range, adjust it/them within its/their setting range.

Chapter 7. PROGRAMMING

7.1 A Program for Controlling an Electric Furnace (with Applying the A/D Conversion Module, PID Control Module and D/A Conversion Module)

1) System Configuration



2) Initial Settings

(1) PID control module
- A) Used loop : loop 0
- B) Specifying forward/reverse action : forward action
- C) Setting SV: 12800
- D) Specifying auto/manual processing : auto processing

(2) A/D conversion module
- A) Used channel: channel 0
- B) Specifying output data type: -192 to 16191
- C) Setting filter constant: 50

(3) D/A conversion module
- A) Used channel: channel 0
- B) Specifying input data type: -192 to 16191
- C) The output when no channel is used or the CPU module is in the stop state : The median value of the output will be output.

3) Descriptions of the Program

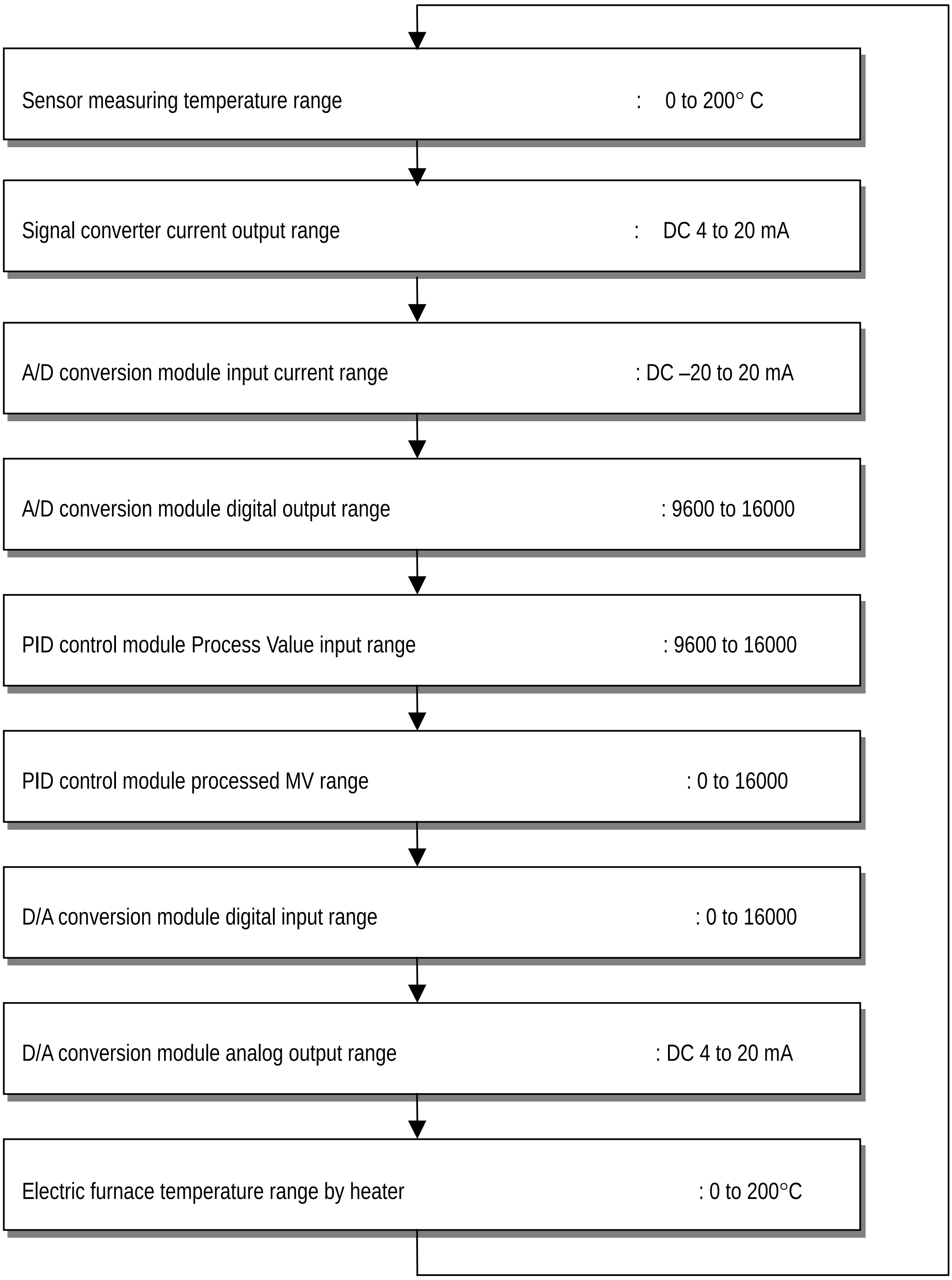
(1) A temperature 0 to 200°C from the temperature sensor is converted into an analog signal 4 to 20 mA and then the signal is input to the channel 0 of the A/D conversion module channel and converted into a digital value 9600 to 16000.

(2) In the PID control module, 100°C (where the signal converter output is 12 mA and 12800 as a digital value.) is set as SV. With regards to P.I.D constants, the manipulated value in the BCD digital switch is set to the proportional constant when %I0.1.0 is on, to the integral constant when %I0.1.1 is on, and to the derivative constant when %I0.1.2 is on.

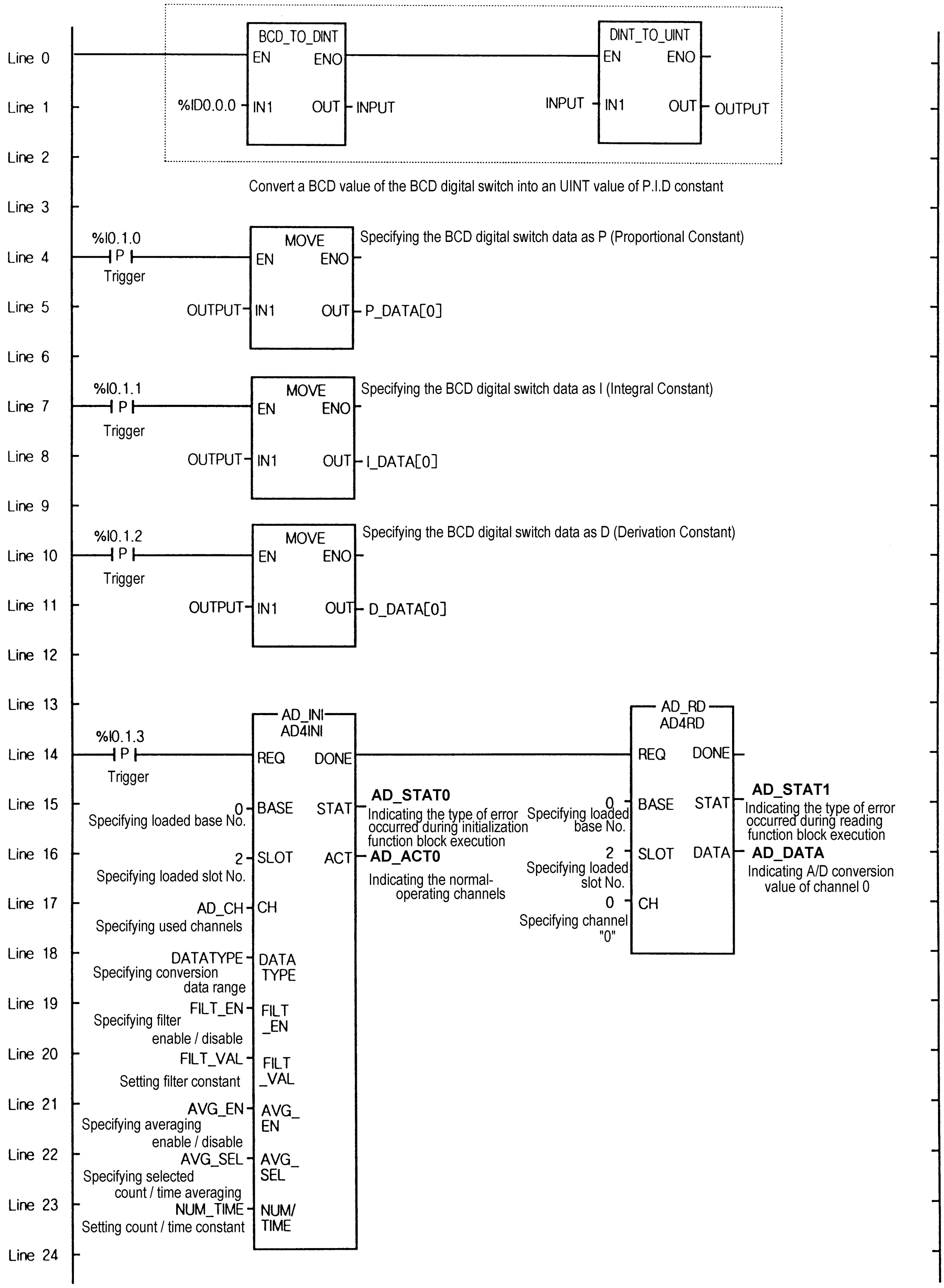
(3) MV, the result from PID processing is output at the channel 0 of the D/A conversion module.

(4) If %I0.1.3 turns on, initial setting of the A/D conversion module, PID control module and D/A conversion module is executed.

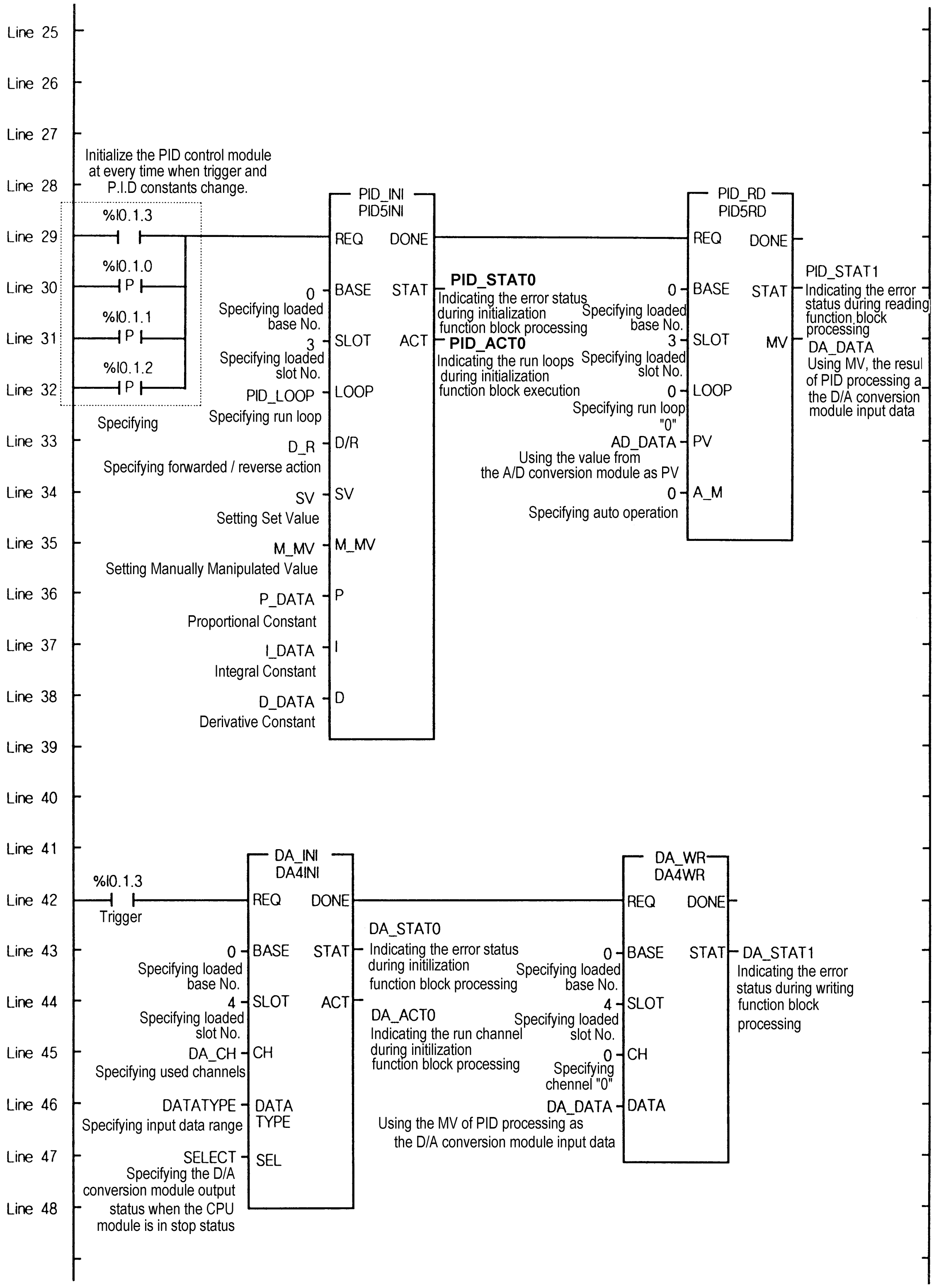
1) Modules and their Signal Processing



5) Program



Line 0
Line 1
Line 2
Line 3
Line 4
Line 5
Line 6
Line 7
Line 8
Line 9
Line 10
Line 11
Line 12
Line 13
Line 14
Line 15
Line 16
Line 17
Line 18
Line 19
Line 20
Line 21
Line 22
Line 23
Line 24
BCD_TO_DINT
EN ENO
%ID0.0.0 IN1 OUT INPUT
DINT_TO_UINT
EN ENO
INPUT IN1 OUT OUTPUT
Convert a BCD value of the BCD digital switch into an UINT value of P.I.D constant
%I0.1.0
P
Trigger
MOVE
EN ENO
OUTPUT IN1 OUT P_DATA[0]
Specifying the BCD digital switch data as P (Proportional Constant)
%I0.1.1
P
Trigger
MOVE
EN ENO
OUTPUT IN1 OUT I_DATA[0]
Specifying the BCD digital switch data as I (Integral Constant)
%I0.1.2
P
Trigger
MOVE
EN ENO
OUTPUT IN1 OUT D_DATA[0]
Specifying the BCD digital switch data as D (Derivation Constant)
%I0.1.3
P
Trigger
AD_INI
AD4INI
REQ DONE
0 BASE STAT
Specifying loaded base No.
2 SLOT ACT
Specifying loaded slot No.
AD_CH CH
Specifying used channels
DATATYPE DATA TYPE
Specifying conversion data range
FILT_EN FILT _EN
Specifying filter enable / disable
FILT_VAL FILT _VAL
Setting filter constant
AVG_EN AVG_ EN
Specifying averaging enable / disable
AVG_SEL AVG_ SEL
Specifying selected count / time averaging
NUM_TIME NUM/ TIME
Setting count / time constant
AD_STAT0
Indicating the type of error occurred during initialization function block execution
AD_ACT0
Indicating the normal-operating channels
AD_RD
AD4RD
REQ DONE
0 BASE STAT
Specifying loaded base No.
2 SLOT DATA
Specifying loaded slot No.
0 CH
Specifying channel "0"
AD_STAT1
Indicating the type of error occurred during reading function block execution
AD_DATA
Indicating A/D conversion value of channel 0



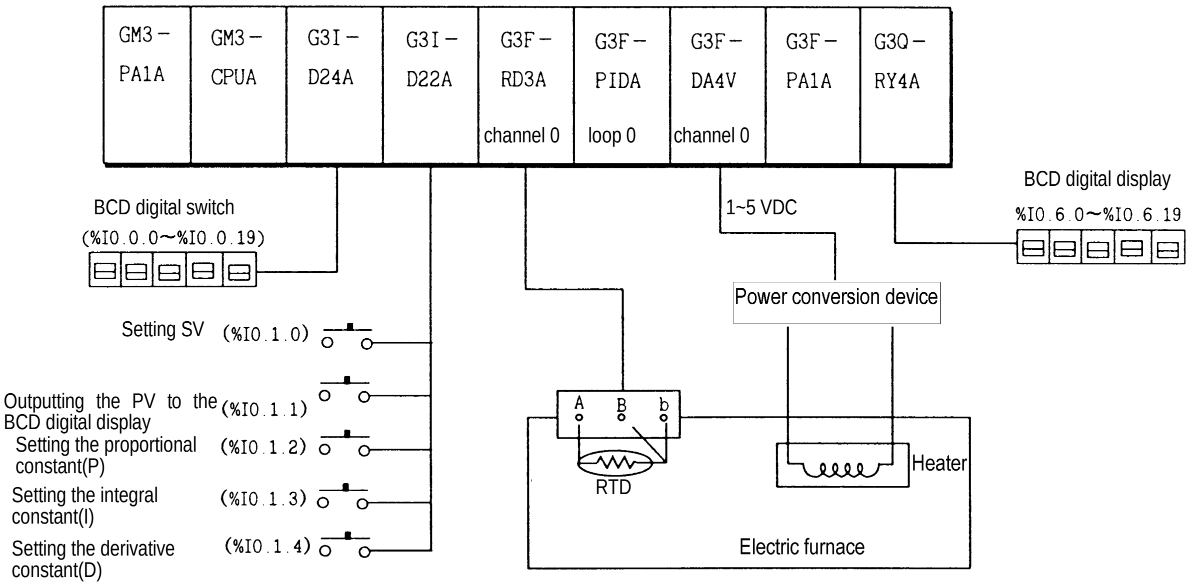
Line 25
Line 26
Line 27
Initialize the PID control module at every time when trigger and P.I.D constants change.
Line 28
PID_INI
PID5INI
%I0.1.3
Line 29
REQ DONE
%I0.1.0
P
Line 30
0 BASE STAT
Specifying loaded base No.
PID_STAT0
Indicating the error status during initialization function block processing
%I0.1.1
P
Line 31
3 SLOT ACT
Specifying loaded slot No.
PID_ACT0
Indicating the run loops during initialization function block execution
%I0.1.2
P
Line 32
PID_LOOP LOOP
Specifying
Specifying run loop
Line 33
D_R D/R
Specifying forwarded / reverse action
Line 34
SV SV
Setting Set Value
Line 35
M_MV M_MV
Setting Manually Manipulated Value
Line 36
P_DATA P
Proportional Constant
Line 37
I_DATA I
Integral Constant
Line 38
D_DATA D
Derivative Constant
PID_RD
PID5RD
REQ DONE
0 BASE STAT
Specifying loaded base No.
3 SLOT MV
Specifying loaded slot No.
0 LOOP
Specifying run loop "0"
AD_DATA PV
Using the value from the A/D conversion module as PV
0 A_M
Specifying auto operation
PID_STAT1
Indicating the error status during reading function block processing
DA_DATA
Using MV, the resul of PID processing a the D/A conversion module input data
Line 39
Line 40
Line 41
DA_INI
DA4INI
%I0.1.3
Line 42
Trigger
REQ DONE
Line 43
0 BASE STAT
Specifying loaded base No.
DA_STAT0
Indicating the error status during initilization function block processing
Line 44
4 SLOT ACT
Specifying loaded slot No.
DA_ACT0
Indicating the run channel during initilization function block processing
Line 45
DA_CH CH
Specifying used channels
Line 46
DATATYPE DATA TYPE
Specifying input data range
Line 47
SELECT SEL
Specifying the D/A conversion module output status when the CPU module is in stop status
Line 48
DA_WR
DA4WR
REQ DONE
0 BASE STAT
Specifying loaded base No.
4 SLOT
Specifying loaded slot No.
0 CH
Specifying chennel "0"
DA_DATA DATA
Using the MV of PID processing as the D/A conversion module input data
DA_STAT1
Indicating the error status during writing function block processing

6) I/O Variables Used in the Program

Variable Name	Var_Kind	Data Type	(AT Address) (Initial Value)
AD_ACT0	: VAR	: ARRAY [0..15] OF BOOL	
AD_CH	: VAR	: ARRAY [0..15] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
AD_DATA	: VAR	: INT	
AD_INI	: VAR	: FB Instance	
AD_RD	: VAR	: FB Instance	
AD_STAT0	: VAR	: USINT	
AD_STAT1	: VAR	: USINT	
AVG_EN	: VAR	: ARRAY [0..3] OF BOOL	: = { 0.0.0.0 }
AVG_SEL	: VAR	: ARRAY [0..3] OF BOOL	: = { 0.0.0.0 }
D_DATA	: VAR	: ARRAY [0..31] OF UINT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
D_R	: VAR	: ARRAY [0..31] OF BOOL	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
DA_ACT0	: VAR	: ARRAY [0..15] OF BOOL	
DA_CH	: VAR	: ARRAY [0..15] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
DA_DATA	: VAR	: INT	
DA_INI	: VAR	: FB Instance	
DA_STAT0	: VAR	: USINT	
DA_STAT1	: VAR	: USINT	
DA_WR	: VAR	: FB Instance	
DATATYPE	: VAR	: ARRAY [0..15] OF BOOL	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
FILT_EN	: VAR	: ARRAY [0..3] OF BOOL	: = { 1.0.0.0 }
FILT_VAL	: VAR	: ARRAY [0..3] OF USINT	: = { 50.0.0.0 }
I_DATA	: VAR	: ARRAY [0..31] OF UINT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
INPUT	: VAR	: DINT	
M_MV	: VAR	: ARRAY [0..31] OF INT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
NUM_TIME	: VAR	: ARRAY [0..3] OF UINT	: = { 0.0.0.0 }
OUTPUT	: VAR	: UINT	
P_DATA	: VAR	: ARRAY [0..31] OF UINT	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
PID_ACT0	: VAR	: ARRAY [0..31] OF BOOL	
PID_INI	: VAR	: FB Instance	
PID_LOOP	: VAR	: ARRAY [0..31] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
PID_RD	: VAR	: FB Instance	
PID_STAT0	: VAR	: USINT	
PID_STAT1	: VAR	: USINT	
SELECT	: VAR	: ARRAY [0..15] OF USINT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
SV	: VAR	: ARRAY [0..31] OF INT	: = { 12800.0.0.0.0.0.0.0.0. 0.0.0.0.0.0.0.0.0.0. 0.0.0.0.0.0.0.0.0.0. 0.0.0 }

7.2 A Program for Control Using a RTD (with Applying the RTD Input Module, PID Control Module and D/A Conversion Module)

1) System Configuration



2) Initial Settings

(1) PID control module
- A) Specifying used loop : loop 0
- B) Specifying forward/reverse action: forward action
- C) Specifying auto/manual processing : auto processing

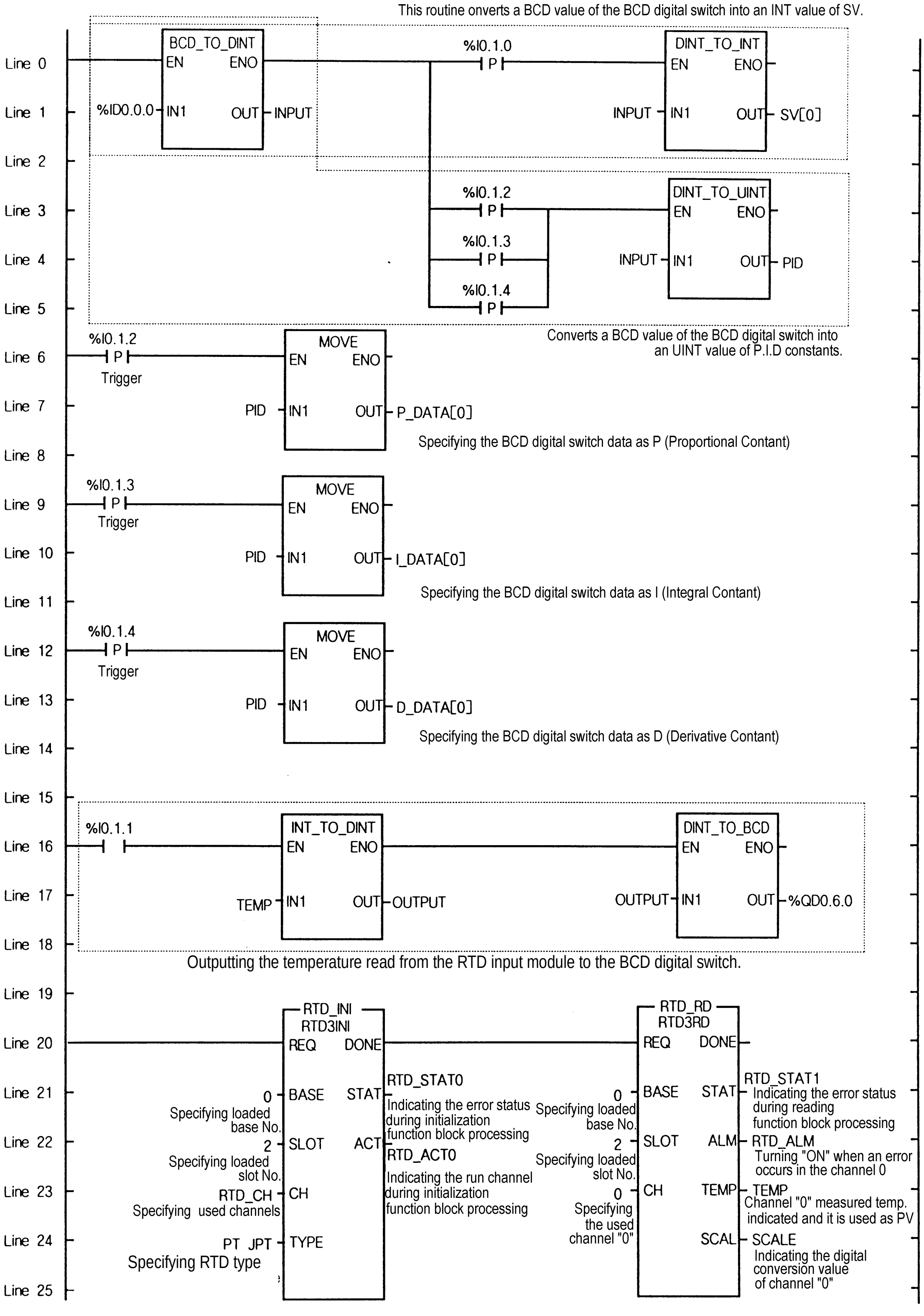
(2) RTD input module
- A) Specifying used channel: channel 0
- B) Specifying RTD sensor type: Pt100

(3) D/A conversion module
- A) Setting the voltage input range to –5 to 5 VDC (offset: 1 VDC, gain: 3 VDC)
- B) Specifying used channel : channel 0
- C) Specifying input data type : 0 to 16000

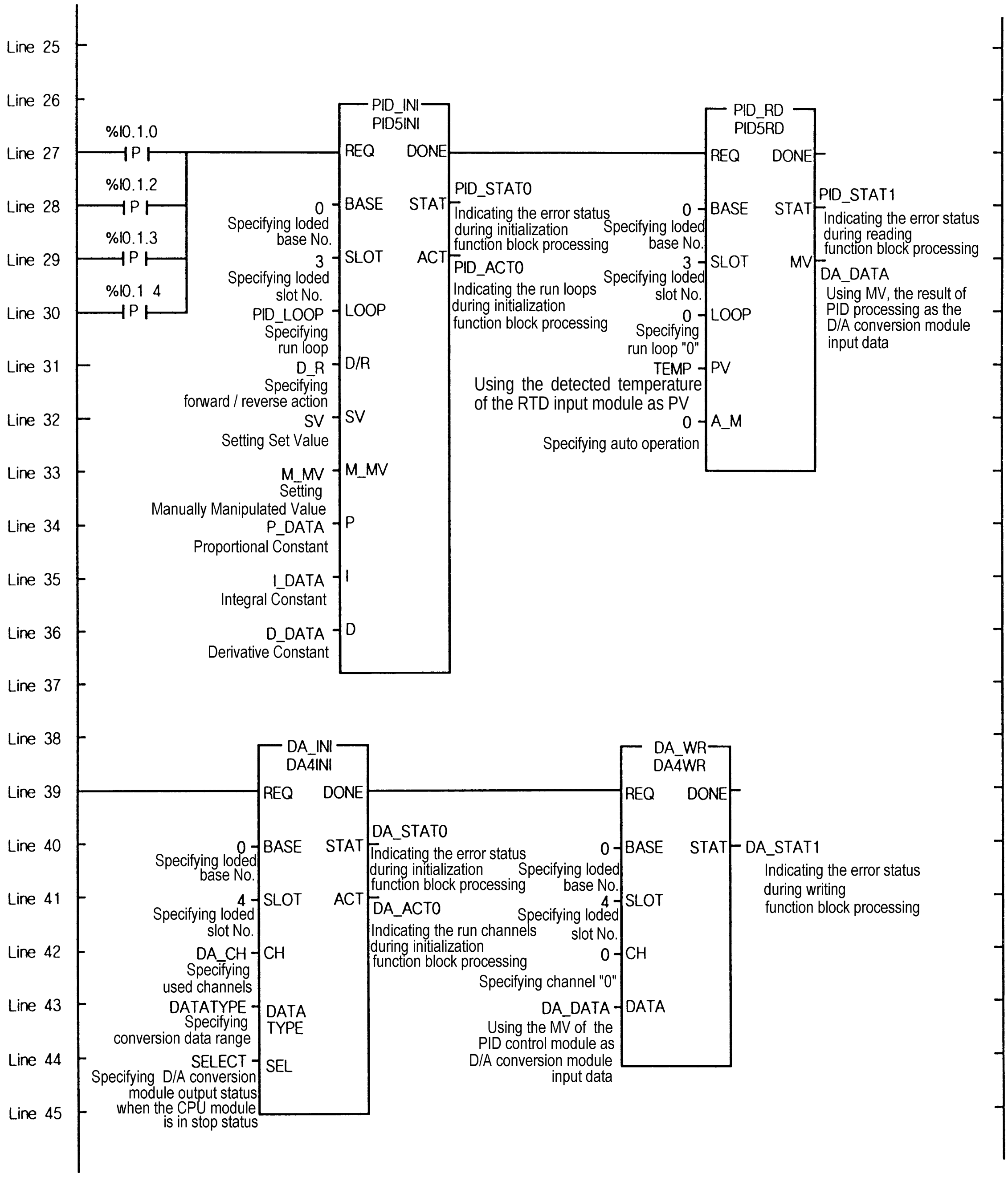
3) Descriptions of the Program

(1) The channel 0 of the RTD input module detects a temperature of the electric furnace through Pt100 and receives it as a digital value.
(2) The Set Value of PID control module loop 0 is set to 8000(where the temperature is 100C). With regards to P.I.D constants, the manipulated value in the BCD digital switch is set to the proportional constant when %I0.1.2 is turned on, as the integral constant when %I0.1.3 is turned on, and as the derivative constant when %I0.1.4 is turned on.
(3) As the change of MV, the manipulated value in the BCD digital switch is set to a new MV when %I0.1.0 is turned on.
(4) MV, the result from PID processing is output at the channel 0 of the D/A conversion module.
(5) If %I0.1.1 turns on, PV is displayed on the BCD digital display.

4) Program



This routine onverts a BCD value of the BCD digital switch into an INT value of SV.
BCD_TO_DINT
EN ENO
%ID0.0.0 IN1 OUT INPUT
%I0.1.0
DINT_TO_INT
INPUT IN1 OUT SV[0]
%I0.1.2
DINT_TO_UINT
%I0.1.3
INPUT IN1 OUT PID
%I0.1.4
Converts a BCD value of the BCD digital switch into an UINT value of P.I.D constants.
%I0.1.2
Trigger
MOVE
PID IN1 OUT P_DATA[0]
Specifying the BCD digital switch data as P (Proportional Contant)
%I0.1.3
Trigger
MOVE
PID IN1 OUT I_DATA[0]
Specifying the BCD digital switch data as I (Integral Contant)
%I0.1.4
Trigger
MOVE
PID IN1 OUT D_DATA[0]
Specifying the BCD digital switch data as D (Derivative Contant)
%I0.1.1
INT_TO_DINT
TEMP IN1 OUT OUTPUT
DINT_TO_BCD
OUTPUT IN1 OUT %QD0.6.0
Outputting the temperature read from the RTD input module to the BCD digital switch.
RTD_INI
RTD3INI
REQ DONE
0 BASE STAT RTD_STAT0
Specifying loaded base No.
Indicating the error status during initialization function block processing
2 SLOT ACT RTD_ACT0
Specifying loaded slot No.
Indicating the run channel during initialization function block processing
RTD_CH CH
Specifying used channels
PT JPT TYPE
Specifying RTD type
RTD_RD
RTD3RD
REQ DONE
0 BASE STAT RTD_STAT1
Specifying loaded base No.
Indicating the error status during reading function block processing
2 SLOT ALM RTD_ALM
Specifying loaded slot No.
Turning "ON" when an error occurs in the channel 0
0 CH TEMP TEMP
Specifying the used channel "0"
Channel "0" measured temp. indicated and it is used as PV
SCAL SCALE
Indicating the digital conversion value of channel "0"

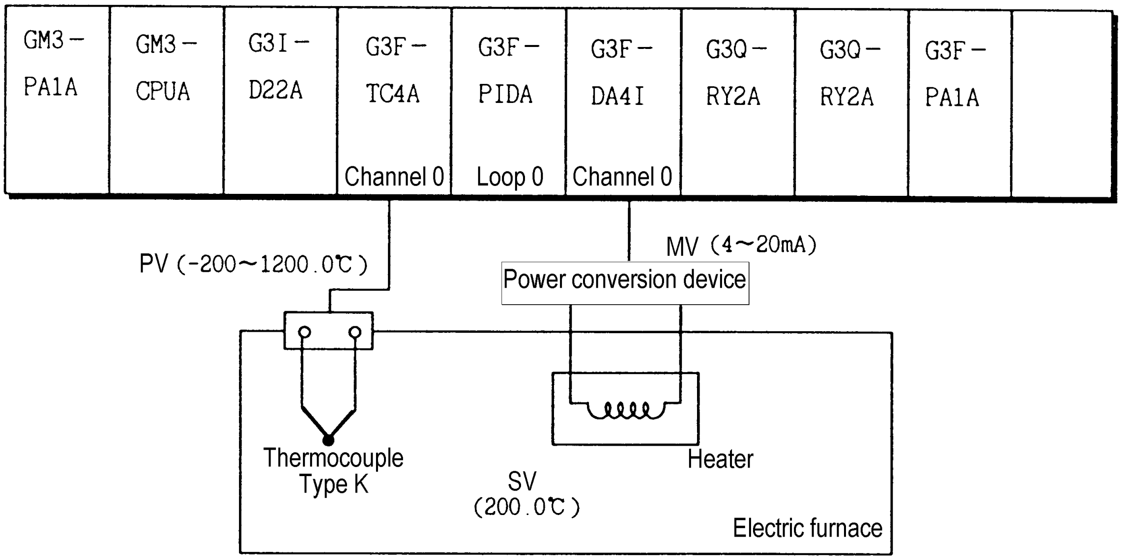

Line 25
Line 26
Line 27
Line 28
Line 29
Line 30
Line 31
Line 32
Line 33
Line 34
Line 35
Line 36
Line 37
Line 38
Line 39
Line 40
Line 41
Line 42
Line 43
Line 44
Line 45
%I0.1.0
%I0.1.2
%I0.1.3
%I0.1 4
P
PID_INI
PID5INI
REQ
DONE
BASE
STAT
SLOT
ACT
LOOP
D/R
SV
M_MV
P
I
D
0
Specifying loded base No.
3
Specifying loded slot No.
PID_LOOP
Specifying run loop
D_R
Specifying forward / reverse action
SV
Setting Set Value
M_MV
Setting Manually Manipulated Value
P_DATA
Proportional Constant
I_DATA
Integral Constant
D_DATA
Derivative Constant
PID_STAT0
Indicating the error status during initialization function block processing
PID_ACT0
Indicating the run loops during initialization function block processing
PID_RD
PID5RD
REQ
DONE
BASE
STAT
SLOT
MV
LOOP
PV
A_M
0
Specifying loded base No.
3
Specifying loded slot No.
0
Specifying run loop "0"
TEMP
Using the detected temperature of the RTD input module as PV
0
Specifying auto operation
PID_STAT1
Indicating the error status during reading function block processing
DA_DATA
Using MV, the result of PID processing as the D/A conversion module input data
DA_INI
DA4INI
REQ
DONE
BASE
STAT
SLOT
ACT
CH
DATA TYPE
SEL
0
Specifying loded base No.
4
Specifying loded slot No.
DA_CH
Specifying used channels
DATATYPE
Specifying conversion data range
SELECT
Specifying D/A conversion module output status when the CPU module is in stop status
DA_STAT0
Indicating the error status during initialization function block processing
DA_ACT0
Indicating the run channels during initialization function block processing
DA_WR
DA4WR
REQ
DONE
BASE
STAT
SLOT
CH
DATA
0
Specifying loded base No.
4
Specifying loded slot No.
0
Specifying channel "0"
DA_DATA
Using the MV of the PID control module as D/A conversion module input data
DA_STAT1
Indicating the error status during writing function block processing

5) I/O Variables Used in the Program

Variable Name	Var_Kind	Data Type	(AT Address) (Initial Value)
D_DATA	: VAR	: ARRAY [0..31] OF UINT	
D_R	: VAR	: ARRAY [0..31] OF BOOL	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
DA_ACTO	: VAR	: ARRAY [0..15] OF BOOL	
DA_CH	: VAR	: ARRAY [0..15] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
DA_DATA	: VAR	: INT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
DA_INI	: VAR	: FB Instance	
DA_STAT0	: VAR	: USINT	
DA_STAT1	: VAR	: USINT	
DA_WR	: VAR	: FB Instance	
DATATYPE	: VAR	: ARRAY [0..15] OF BOOL	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
I_DATA	: VAR	: ARRAY [0..31] OF UINT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
INPUT	: VAR	: DINT	
M_MV	: VAR	: ARRAY [0..31] OF INT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
OUTPUT	: VAR	: UINT	
P_DATA	: VAR	: ARRAY [0..31] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
PID	: VAR	: UINT	
PID_ACTO	: VAR	: ARRAY [0..31] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
PID_INI	: VAR	: FB Instance	
PID_LOOP	: VAR	: ARRAY [0..31] OF BOOL	
PID_RD	: VAR	: FB Instance	
PID_STAT0	: VAR	: USINT	
PID_STAT1	: VAR	: USINT	
PT_JPT	: VAR	: ARRAY [0..7] OF BOOL	: = { 0.0.0.0.0.0.0.0 }
RTD_ACTO	: VAR	: ARRAY [0..7] OF BOOL	: = { 1.0.0.0.0.0.0.0 }
TRD_ALM	: VAR	: BOO	
RTD_CH	: VAR	: ARRAY [0..7] OF BOOL	
RTD_INI	: VAR	: FB Instance	
RTD_RD	: VAR	: FB Instance	
RTD_STAT0	: VAR	: USINT	
RTD_STAT1	: VAR	: USINT	
SCALE	: VAR	: INT	
SELECT	: VAR	: ARRAY [0..15] OF USINT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
SV	: VAR	: ARRAY [0..31] OF INT	: = { 12800.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
TEMP	: VAR	: INT	

7.3 A Program for Control Using a Thermocouple (with Applying the TC Input module, PID Control Module and D/A Conversion Module)

1) System Configuration



2) Initial Settings

(1) TC input module
  A) Specifying used channel : channel 0
  B) Specifying TC type: K type

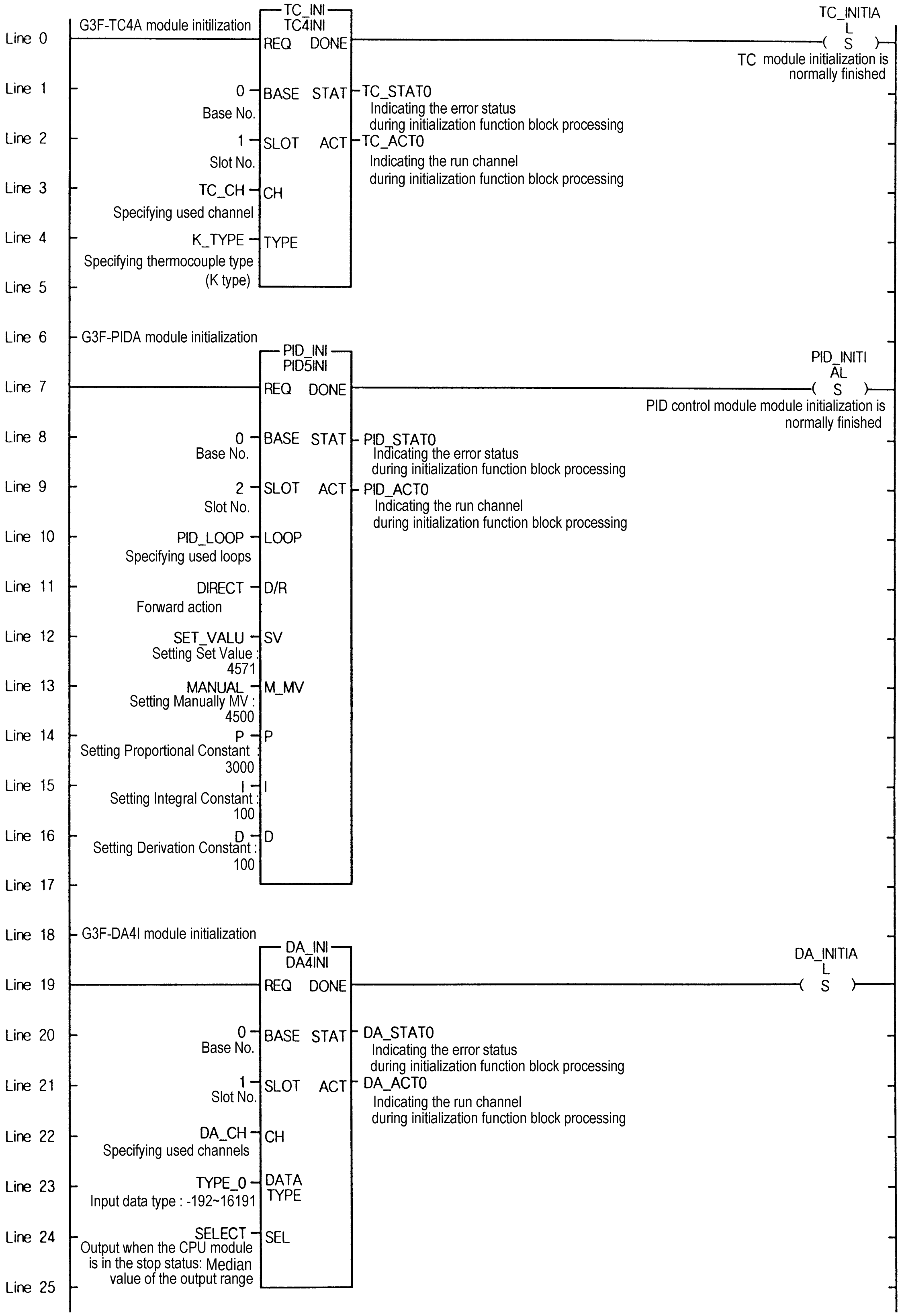
(2) PID control module
  A) Specifying used loop : loop 0
  B) Specifying forward/reverse action: forward action
  C) Specifying auto/manual processing : auto processing
  D) Setting SV: 200°C (4571 as digital value)
  E) Setting M_MV (Used when errors occur) : 4500
  F) Setting P : 3000
  G) Setting I : 100
  H) Setting D : 100
  I) Auto processing is changed to manual processing when errors occur.

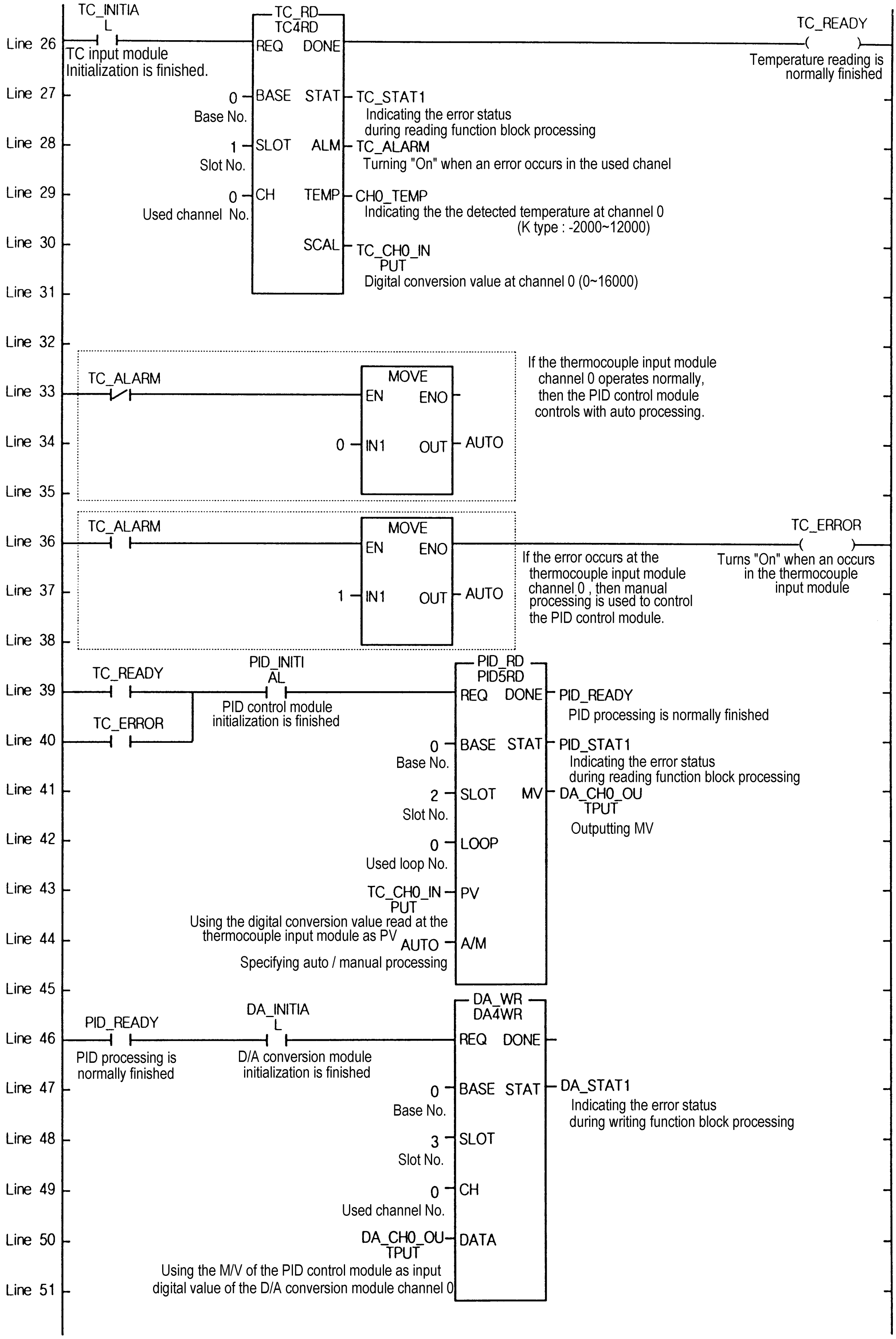
(3) D/A conversion module
  A) Specifying used channel: channel 0
  B) Specifying input data type: -192 to 16191
  C) Output when the CPU module is in the stop state : The median value of the output range is output.

3) Descriptions of the Program

1) The temperature of the electric furnace is converted into a digital value through the channel 0 of the thermocouple input module, and the digital value in the reading function block output variable SCAL is used as PV of the PID control module.
2) The MV of the PID control module is used as input digital data of the channel 0 of the D/A conversion module.
3) If ALM, an output variable of the reading function block of the thermocouple input module turns On, A/M, an input variable of the read function block of the PID control module changes from “0” to “1” and the manual control processing is executed.

4) **Program**


Line 0
G3F-TC4A module initilization
TC_INI
TC4INI
REQ DONE
TC_INITIAL
S
TC module initialization is normally finished
Line 1
0
BASE STAT
TC_STAT0
Base No.
Indicating the error status during initialization function block processing
Line 2
1
SLOT ACT
TC_ACT0
Slot No.
Indicating the run channel during initialization function block processing
Line 3
TC_CH
CH
Specifying used channel
Line 4
K_TYPE
TYPE
Specifying thermocouple type (K type)
Line 5
Line 6
G3F-PIDA module initialization
PID_INI
PID5INI
Line 7
REQ DONE
PID_INITIAL
S
PID control module module initialization is normally finished
Line 8
0
BASE STAT
PID_STAT0
Base No.
Indicating the error status during initialization function block processing
Line 9
2
SLOT ACT
PID_ACT0
Slot No.
Indicating the run channel during initialization function block processing
Line 10
PID_LOOP
LOOP
Specifying used loops
Line 11
DIRECT
D/R
Forward action
Line 12
SET_VALU
SV
Setting Set Value : 4571
Line 13
MANUAL
M_MV
Setting Manually MV : 4500
Line 14
P
P
Setting Proportional Constant : 3000
Line 15
I
I
Setting Integral Constant : 100
Line 16
D
D
Setting Derivation Constant : 100
Line 17
Line 18
G3F-DA4I module initialization
DA_INI
DA4INI
Line 19
REQ DONE
DA_INITIAL
S
Line 20
0
BASE STAT
DA_STAT0
Base No.
Indicating the error status during initialization function block processing
Line 21
1
SLOT ACT
DA_ACT0
Slot No.
Indicating the run channel during initialization function block processing
Line 22
DA_CH
CH
Specifying used channels
Line 23
TYPE_0
DATA TYPE
Input data type : -192~16191
Line 24
SELECT
SEL
Output when the CPU module is in the stop status: Median value of the output range
Line 25


Line 26
TC_INITIAL
TC input module Initialization is finished.
TC_RD
TC4RD
REQ DONE
TC_READY
Temperature reading is normally finished
Line 27
0 BASE STAT TC_STAT1
Base No.
Indicating the error status during reading function block processing
Line 28
1 SLOT ALM TC_ALARM
Slot No.
Turning "On" when an error occurs in the used chanel
Line 29
0 CH TEMP CH0_TEMP
Used channel No.
Indicating the the detected temperature at channel 0 (K type : -2000~12000)
Line 30
SCAL TC_CH0_INPUT
Digital conversion value at channel 0 (0~16000)
Line 31
Line 32
Line 33
TC_ALARM
MOVE
EN ENO
If the thermocouple input module channel 0 operates normally, then the PID control module controls with auto processing.
Line 34
0 IN1 OUT AUTO
Line 35
Line 36
TC_ALARM
MOVE
EN ENO
TC_ERROR
If the error occurs at the thermocouple input module channel 0 , then manual processing is used to control the PID control module.
Turns "On" when an occurs in the thermocouple input module
Line 37
1 IN1 OUT AUTO
Line 38
Line 39
TC_READY
PID_INITIAL
PID control module initialization is finished
PID_RD
PID5RD
REQ DONE PID_READY
PID processing is normally finished
Line 40
TC_ERROR
0 BASE STAT PID_STAT1
Base No.
Indicating the error status during reading function block processing
Line 41
2 SLOT MV DA_CH0_OUTPUT
Slot No.
Outputting MV
Line 42
0 LOOP
Used loop No.
Line 43
TC_CH0_INPUT PV
Using the digital conversion value read at the thermocouple input module as PV
Line 44
AUTO A/M
Specifying auto / manual processing
Line 45
Line 46
PID_READY
PID processing is normally finished
DA_INITIAL
D/A conversion module initialization is finished
DA_WR
DA4WR
REQ DONE
Line 47
0 BASE STAT DA_STAT1
Base No.
Indicating the error status during writing function block processing
Line 48
3 SLOT
Slot No.
Line 49
0 CH
Used channel No.
Line 50
DA_CH0_OUTPUT DATA
Using the M/V of the PID control module as input digital value of the D/A conversion module channel 0
Line 51

5) I/O Variables Used in the Program

Variable Name	Var_Kind	Data Type	(AT Address) (Initial Value)
AUTO	: VAR	: BOOL	
CHO_TEMP	: VAR	: INT	
D	: VAR	: ARRAY [0..31] OF UINT	: = { 100.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
DA_ACT0	: VAR	: ARRAY [0..15] OF BOOL	
DA_CH	: VAR	: ARRAY [0..15] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
DA_CH0_OUTPUT	: VAR	: INT	
DA_INI	: VAR	: FB Instance	
DA_INITIAL	: VAR	: BOOL	
DA_STAT0	: VAR	: USINT	
DA_STAT1	: VAR	: USINT	
DA_WR	: VAR	: FB Instance	
DIRECT	: VAR	: ARRAY [0..31] OF BOOL	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
I	: VAR	: ARRAY [0..31] OF UINT	: = { 100.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
K_TYPE	: VAR	: ARRAY [0..15] OF USINT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
MANUAL	: VAR	: ARRAY [0..31] OF INT	: = { 4500.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
P	: VAR	: ARRAY [0..31] OF UINT	: = { 3000.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
PID_ACT0	: VAR	: ARRAY [0..31] OF BOOL	
PID_INI	: VAR	: FB Instance	
PID_INITIAL	: VAR	: BOOL	
PID_LOOP	: VAR	: ARRAY [0..31] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
PID_RD	: VAR	: FB Instance	
PID_READY	: VAR	: BOOL	
PID_STAT0	: VAR	: USINT	
PID_STAT1	: VAR	: USINT	
SELECT	: VAR	: ARRAY [0..15] OF USINT	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
SET_VALU	: VAR	: ARRAY [0..31] OF INT	: = { 4571.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
TC_ACT0	: VAR	: ARRAY [0..15] OF BOOL	
TC_ALARM	: VAR	: BOOL	
TC_CH	: VAR	: ARRAY [0..15] OF BOOL	: = { 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }
TC_CH0_INPUT	: VAR	: INT	
TC_ERROR	: VAR	: BOOL	
TC_INI	: VAR	: FB Instance	
TC_INITIAL	: VAR	: BOOL	
TC_RD	: VAR	: FB Instance	
TC_READY	: VAR	: BOOL	
TC_STAT0	: VAR	: USINT	
TC_STAT1	: VAR	: USINT	
TYPE_0	: VAR	: ARRAY [0..15] OF BOOL	: = { 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 }