

1. PID CONTROL

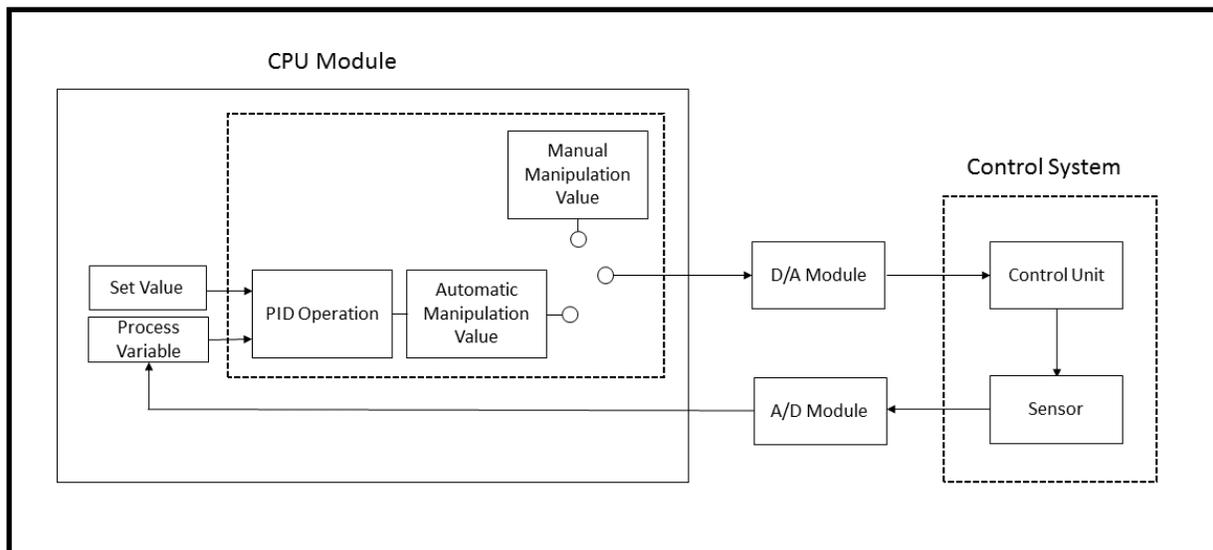
1.1 What is PID Control?

PID control is a control loop feedback mechanism in which the controller continuously calculates an “error value” as the difference between a measured process variable (PV) and a desired set point.

This **Proportional**, **Integral** and **Derivative** controller attempts to minimize the error over time by adjustment of a control variable to a new value. In this control, **P** accounts for present values of the error, **I** accounts for past value of the error, and **D** accounts for predicted future values of the error, based on its current rate of change.

1.2 PID Process

The PID controller calculates the manipulation value (MV) based on the previously set value (SV) and the process variable (PV) from the A/D conversion module. Then, the calculated manipulation value is written on the D/A conversion module to be outputted to the control unit.

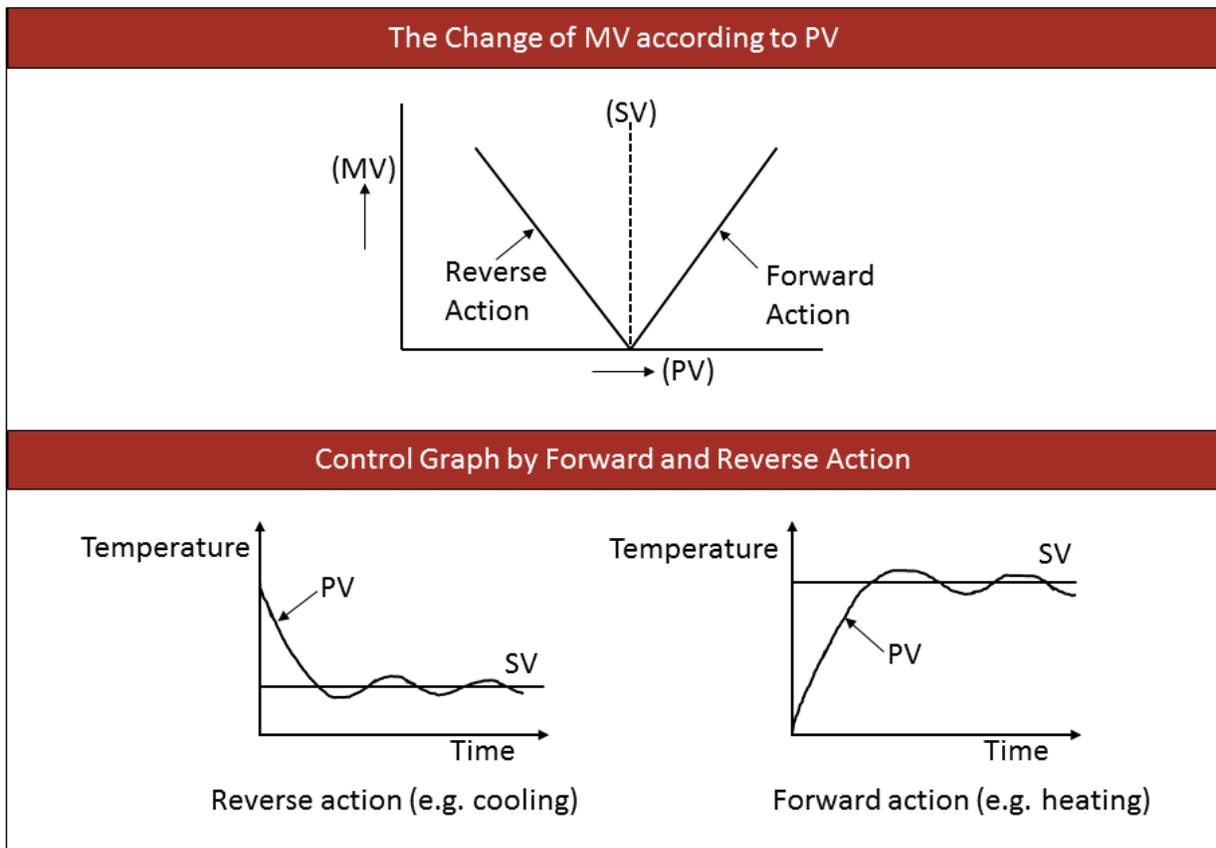


1.3 PID Basics

1.3.1 Forward action and Reverse action

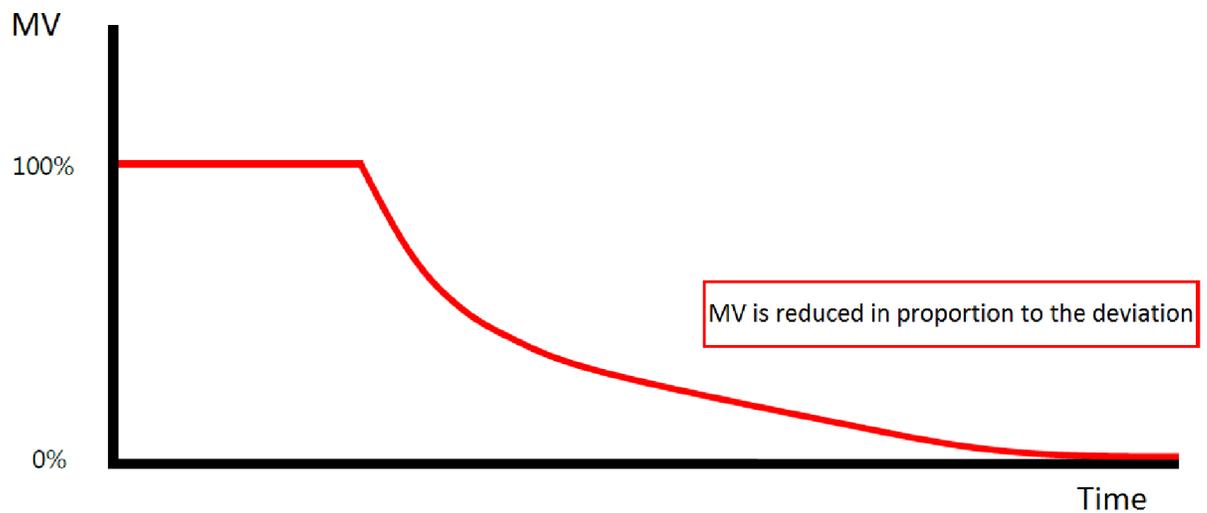
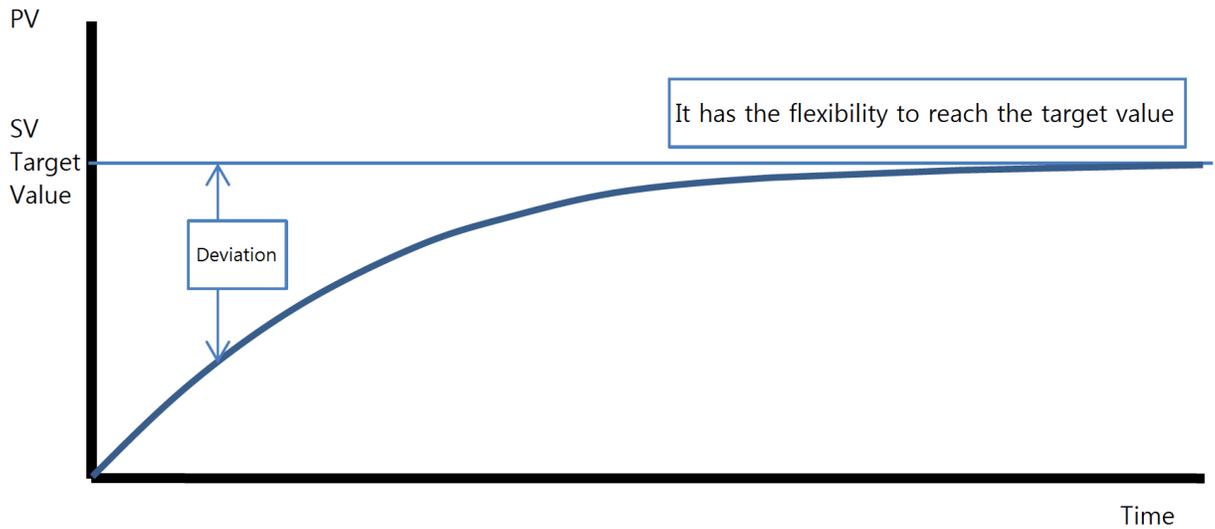
Forward action: When the PV is higher than the SV, it will increase the manipulation value.

Reverse action: When the PV is lower than the SV, it will increase the manipulation value.



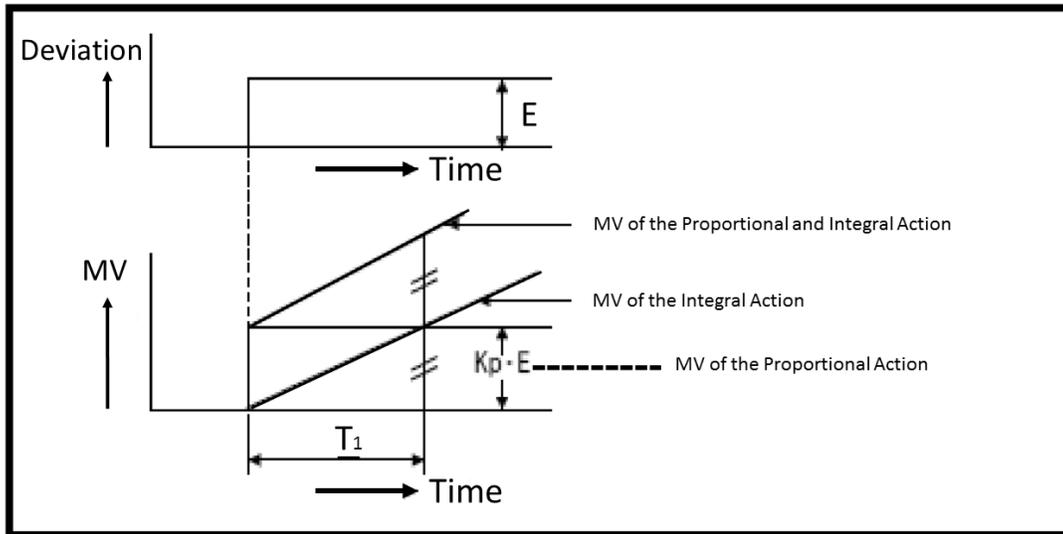
1.3.2 Proportional Control

Proportional control (P) generates the MV in proportion to the error (difference between SV and PV). The manipulation value (MV) is proportional in size to the difference between the target value and current position. Through the proportional control, it is possible to get the MV close to the target value.



1.3.3 Integral Control

Integral control (I) continuously changes the MV in proportion to the time-integral of the error so as to eliminate the deviation. Using the integral action, a user can eliminate the offset (deviation) caused by proportional action.



Integral Action with a Constant Deviation

Integral action is used with PI (Proportional + Integral) control or PID (Proportional + Integral + Derivative) control. The Integral Control cannot be used by itself.

It can be thought that proportional control alone is enough, but in reality when the controlled variable approaches the target value (SV), the manipulated variable (MV) becomes smaller and disables it to control precisely.

As a result, there is this slight difference between the controlled variable (PV) and the target value (SV) over time, which is called “**residual deviation.**” In order to overcome this slight difference, the Integral Control is used to eliminate this residual deviation (offset).

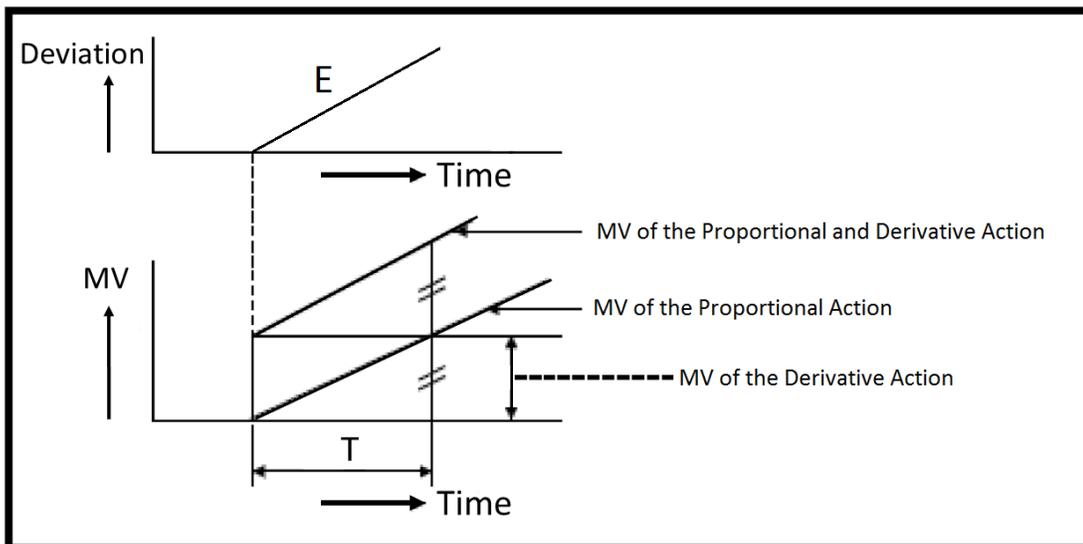
Likewise, when the Proportional action is added with Integral action, it is called “**PI Control.**”

1.3.4 Derivative Control

Derivative (D) control generates the MV in proportion to the rate of change in the error so as to eliminate the deviation (offset). By adding the Derivative control, quick corrective action can be obtained at the beginning of upset condition.

After the deviation occurs in the derivative action, the amount of time it takes for the MV of the derivative action becomes the MV of the proportional action is called the “**derivative time.**”

If the derivative time is set too long, oscillations will occur and the control loop will run unstable. If the derivative time is set 0, the derivative control does not function properly.



Derivative Action when the Deviation Increases at a Constant Rate

The Derivative Control is used with PI (Proportional + Integral) control or PID (Proportional + Integral + Derivative) control. The Derivative Action is never used alone.

1.3.5 PID Control Calculation

Name	Description
En	Current Sampling Deviation
En-1	Deviation of the Preceding Interval
Kp	Proportional Gain
Ki	Integral Gain
Kd	Derivative Gain
Ts	Sampling Interval
α	Filter Coefficient
MVn	Current Manipulation Value
MVn-1	Manipulation Value of the Preceding Sampling Cycle
SV	Set Value
PVn	Process Value of the Present Sampling Cycle
PVnf	Process Value of the Present Sampling Cycle After Filtering

$$MVn = MVn-1 + Kp\{(En-En-1) + (Ts/Ki)*En + (Kd/Ts)*(2PVnf-1-PVnf-PVnf-2)\} \text{ (Direct Action)}$$

$$MVn = MVn-1 + Kp\{(En-En-1) + (Ts/Ki)*En - (Kd/Ts)*(2PVnf-1-PVnf-PVnf-2)\} \text{ (Reverse Action)}$$

$$PVnf = PVn + \alpha(PVnf-1 - PVn)$$

$$En = PVnf - SV \text{ (Reverse Action)}$$

$$En = SV - PVnf \text{ (Direct Action)}$$

1.4 PID Instructions

1.4.1 PID Specifications

Item		Specification	Remarks
Range to Set PID Constant	Proportional Gain (K _p)	1 – 30000	1 – 6553500 (PID_2)
	Integral Gain (K _i)	0 – 30000 (0.0s – 3000.0s)	0 : No Integral Operation
	Derivative Gain (K _d)	0 – 30000 (0.00s – 300.00s)	0 : No Derivative Operation
Sampling Interval (T)		1 – 6000 (0.01s – 60.00s)	
Range of Set Value(SV)		0 – 16,000	
Range of Process Value(PV)		0 – 16,000	
Range of Manipulated Output Value(MV)		0 – 16,000	
Range of Manually Manipulated Value		0 – 16,000	
PID Control Loop		32 Loop	
Control Operation		Forward/Reverse Operation	
Type of Operation		Differentiates measured values	

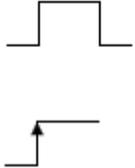
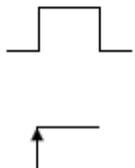
Proportional gain can maximize its setting range up to 100 times by using the function selection flag of the PIDCAL command. By enabling this function, the range value of K_p can be expanded from 1 to a maximum of 6,553,500.

	XP	CP	BP	PLC-S	CICON
Version	V4.27	V4.26	V1.31	V1.37	V3.10

Compared to the preceding versions of PID control, accuracy has been improved for the PID_2 functions and the Auto Tuning algorithm has been modified so that it can now be implemented to a wider range of applications. To obtain an enhanced effect of reaching the target value more quickly, the self-learning function has also been added for the PID_2 functions. These new functions are built into CPU modules, which are produced since the year of 2012, and available in the firmware versions described in the table above.

1.4.2 PID Instructions

*There is no need to manually write the instructions explained in the below table since they are automatically created by the PID Special Program. But if you want to write your own program using the ladder diagram, please refer to the below symbols to enter the PID instructions.

Type	Instruction	Symbol	Processing Information	Condition																							
PID Initialization	PIDINIT	<p>PIDINIT is used to check the items and to initialize the devices for PID operation.</p>  <p>PID control data get assigned from the specified word device address "S."</p>	<table border="0"> <tr> <td rowspan="2">Common Data Setting Area</td> <td rowspan="2">}</td> <td>S + 0</td> </tr> <tr> <td>S + 1</td> </tr> <tr> <td rowspan="2">For Loop 1</td> <td rowspan="2">}</td> <td>S + 2</td> </tr> <tr> <td>∞</td> </tr> <tr> <td rowspan="2">For Loop 2</td> <td rowspan="2">}</td> <td>S + 21</td> </tr> <tr> <td>S + 22</td> </tr> <tr> <td rowspan="2">∞</td> <td rowspan="2">}</td> <td>∞</td> </tr> <tr> <td>S + 41</td> </tr> <tr> <td rowspan="2">For Loop 'n' Data</td> <td rowspan="2">}</td> <td>S + ((n-1)*20 + 2)</td> </tr> <tr> <td>∞</td> </tr> <tr> <td></td> <td></td> <td>S + (n*20 + 1)</td> </tr> </table>	Common Data Setting Area	}	S + 0	S + 1	For Loop 1	}	S + 2	∞	For Loop 2	}	S + 21	S + 22	∞	}	∞	S + 41	For Loop 'n' Data	}	S + ((n-1)*20 + 2)	∞			S + (n*20 + 1)	
Common Data Setting Area	}	S + 0																									
		S + 1																									
For Loop 1	}	S + 2																									
		∞																									
For Loop 2	}	S + 21																									
		S + 22																									
∞	}	∞																									
		S + 41																									
For Loop 'n' Data	}	S + ((n-1)*20 + 2)																									
		∞																									
		S + (n*20 + 1)																									
PID Calculation	PIDCAL	 <p>SV and PV get assigned by specified "S" address and after the PID calculation, the result gets saved in MV according to the "S" word device.</p>	<table border="0"> <tr> <td rowspan="2">For Loop 1</td> <td rowspan="2">}</td> <td>S + 0</td> </tr> <tr> <td>∞</td> </tr> <tr> <td rowspan="2">For Loop 2</td> <td rowspan="2">}</td> <td>S + 19</td> </tr> <tr> <td>S + 20</td> </tr> <tr> <td rowspan="2">∞</td> <td rowspan="2">}</td> <td>∞</td> </tr> <tr> <td>S + 39</td> </tr> <tr> <td rowspan="2">For Loop 'n' Data</td> <td rowspan="2">}</td> <td>S + ((n-1)*20)</td> </tr> <tr> <td>∞</td> </tr> <tr> <td></td> <td></td> <td>S + (n*20 - 1)</td> </tr> </table>	For Loop 1	}	S + 0	∞	For Loop 2	}	S + 19	S + 20	∞	}	∞	S + 39	For Loop 'n' Data	}	S + ((n-1)*20)	∞			S + (n*20 - 1)					
For Loop 1	}	S + 0																									
		∞																									
For Loop 2	}	S + 19																									
		S + 20																									
∞	}	∞																									
		S + 39																									
For Loop 'n' Data	}	S + ((n-1)*20)																									
		∞																									
		S + (n*20 - 1)																									

1.4.3 PIDINIT Instruction

The PIDINIT command checks the items and initializes the devices set up for PID operation.

The PID Control Special Program is represented as shown below.

The screenshot shows the PIDINIT.SRC software interface. At the top, there are buttons for 'Save', 'Monitor', 'Auto Tune', 'New Dialog', and 'Help'. Below these are controls for 'Current Value -> Set Value', 'Current Loop', 'Entire Loops', 'LD Convert', 'Trend', and 'Description'. There are also dropdown menus for 'No. of Loop' (Total: 1, Loops/Scan: 1) and 'Start Data of PID Control' (PIDINIT: 0, PIDCAL: 100). The main part of the interface is a table with the following data:

Index	Device	Set Value	Current Value
Path Calc(0:Forward 1:Reverse)	D00002	Forward	
Sampling Time(0.01 - 60 sec)	D00003	10.00	
Kp(0 - 65535)	D00004	8000	
Ki(0.0 - 3000 sec)	D00005	2000.0	
Kd(0.00 - 300 sec)	D00006	0.00	
Filter(0 - 1.00)	D00007	0.20	
MV Low Limit(0 - 16000)	D00008	0	
MV High Limit(0 - 16000)	D00009	16000	
MV Change Rate Limit(0 - 16000)	D00010	16000	
MV Auto-Apply(0:Disabled 1:Enabled)	D00011	Disable	
SV Ramp(0 - 1000 0:Disabled)	D00012	0	
On/Off Time(0.00 - 60.00)	D00014	0.00	
SV(Set Value)	D00100	0	
PV(Process Value)	D00101		
MV(Manipulation Value)	D00102		
PVnt(After Filter)	D00103		
MVman(Mv Manual)	D00104	0	
(0:Auto 1:Manual)	D00105	Auto	
Self Learning Enable(1) / Disable(0)	D00105	Disable	
Kp x 100 (1)	D00105	-	
PID Error Code	D00015	0	
PID Status Code	D00016		

At the bottom of the interface, there are six status indicators with radio buttons:

- PID Loop Initialized
- Auto-tune Processing
- ON/OFF Ctrl Output
- Self Learn Data Ready
- Stable Status
- PID - 2 Control

*When the value of each item is 0, do not use the item.

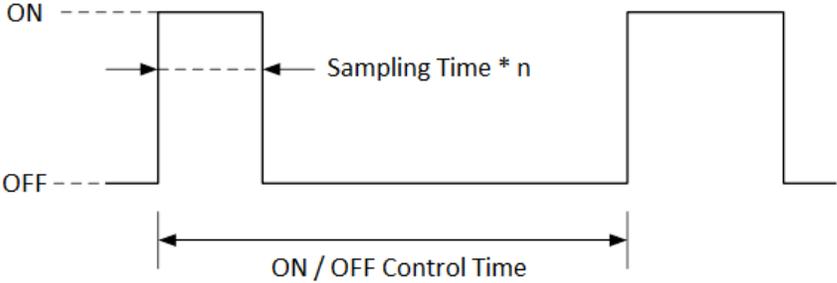
Offset	Item	Description	Range	Remarks
0	Number of Loops		1 – 32	Common for Entire Loops
1	Number of Loops for a Scan		1 – 32	
2(0)	Operation Type	Forward(0) / Reverse(1)	0 / 1	For Loops
3(1)	Sampling Interval (T_s)	0.01 – 60.00s	1 – 6,000	
4(2)	Proportional Gain (K_p)	0.01 – 300.00	1 – 30,000	
5(3)	Integral Gain (K_i)	0.01 – 300.00s	0 – 30,000	
6(4)	Derivative Gain (K_d)	0.00 – 300.00s	0 – 30,000	
7(5)	Filter Coefficient (α)	0.00 – 1.00	0 – 100	
8 (6)	Manipulated Value Low Limit (MVLL)	0 – 16,000	0 – 16,000	
9 (7)	Manipulated Value High Limit (MVHL)	0 – 16,000	0 – 16,000	
10 (8)	Variation Rate Limit of Manipulated Value (Δ MVL)	0 – 16,000	0 – 16,000	
11 (9)	Use of MV Automatic Apply	Not Use(0) / Use(1)	0 / 1	
12 (10)	SV Transition Step	0 – 1,000 times	0 – 1,000	
13 (11)	PV Inertia Amount	0 – 16,000	0 – 16,000	
14 (12)	ON/OFF Control Time	1.00 – 60.00s	1 – 6000 0: Not Use	
15 (13)	PIDINIT Operation Status	Indicates Errors	*Table	
16 (14)	FLAGS			Not Available to Users
17 (15)	Reserved			
18 (16)	Auto Tuning Counter 1			
19 (17)	Auto Tuning Counter 2			
20 (18)	On/OFF Control Counter			
21 (19)	Sampling Counter			

a) Common setting data

Offset	Item	Description
0	Number of Loops	Configures the number of loops to operate PID. To control an object, one PID operation loop is used. Up to 32 loops.
1	Number of Loops for a Scan	Set up the number of loops for one scan in case of several loops at sampling intervals.

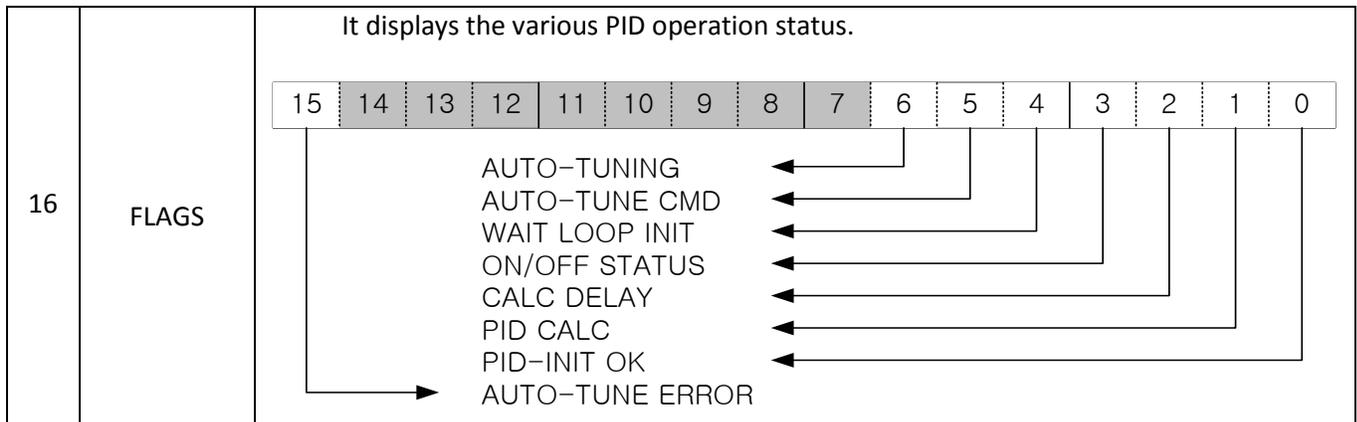
b) Setting data for each loop

Offset	Item	Description
2	Operation Type	Determines Forward (0) or Reverse (1) operation.
3	Sampling Interval (T_s)	This is used to set up PID calculation interval. Generally, this interval should be the interval of PV change while MV output is at maximum status. Typically, approximately 5-second-interval is recommended for a building temperature control as an example.
4	Proportional Gain (K_p)	Configures the proportional gain. The value of K_p can be set from 1 to 65535. Proportional gain can maximize its setting range up to 100 times by using the function selection flag of the PIDCAL command. By enabling this function, the range value of K_p can be expanded from 100 to a maximum of 6,553,500. This function is inclusive of PID_2.
5	Integral Gain (K_i)	Configures the integral gain.
6	Derivative Gain (K_d)	Configures the derivative gain.
7	Filter Coefficient (α)	This is used to set up the range of the filtering effect on the measured value (PV), which is entered from an A/D card. The closer it gets to 0, the less filtering effect is.
8	MV Low Limit	Sets up the lowest manipulated value for the PID operation in auto mode. By enabling this function, a manipulated value (MV) can never be lower than the fixed minimum manipulated value.
9	MV High Limit	Sets up the highest manipulated value for the PID operation in auto mode. By enabling this function, a manipulated value (MV) can never be higher than the fixed maximum manipulated value.
10	Variation Rate Limit of MV (ΔMVL)	Sets the MV change rate limit which compares with the change rate of current manipulated value and the previous manipulated value. When the change of the MV is greater than the fixed MV change rate limit, the output value is outputted as the fixed variation rate limit.
11	Use of MV Automatic Apply	Use (1): MV value gets automatically passed to the manual MV in auto mode. When switched to manual mode, the current MV value can be maintained. Not Use (0): When converted to manual mode, the initial setting value is applied as the manipulated value (MV).
12	SV Transition Step	To control the occurrence of overshoot due to a sharply changed target value (SV) and to keep stable control, this is used to divide a variation value into 10 steps and to increase or decrease the value progressively at sampling intervals (T_s) when SV is changed.

13	PV Inertia Amount	It is the measurement added to the PID_2 version. The PV Inertia amount is detected by measuring the change in PV according to the MV output during the Auto Tuning process. The detected amount of inertia is automatically written here. The PV inertia amount is used for self-learning function to determine when to block the MV output.
14	ON/OFF Control Time	<p>When using the ON/OFF control mode, the user specifies the control reference time here. The control time is specified in units of 10mS from 0 second to 60.00 seconds. When this feature is not in use, specify '0.'</p> <p>According to the PID calculation result, the ON time is determined within the time range. The control output resolution can be calculated by (Sampling time) / (ON and OFF control time). When it is ON state, the value of MV is outputted as the maximum MV. When it is OFF state, the value of MV is outputted as the minimum MV. The recommended ON/OFF control time is sampling interval * 10 or above.</p> 
15	PID Operation Status (Error Code)	Indicates the status of error while PID operation is initialized. *Refer to the table below.

Code	Error Name	Error Description	Range
0	No Error		
1	Error in Number of Loops	When a value is out of the range (1 – 32)	For entire loops
2	Error in 1 Scan Loop No.	Though there is no error in the value, PID operation can be omitted if the setting value is too small.	
3	Error in Operation Type	When a value is neither 0 nor 1	For each loop
4	Error in Sampling Interval (Ts)	When a value is out of the range (1 – 6,000)	
5	Error in Proportional Gain (Kp)	When a value is out of the range (1 – 10,000)	
6	Error in Integral Gain (Ki)	When a value is out of the range (0 – 30,000)	
7	Error in Derivative Gain (Kd)	When a value is out of the range (0 – 30,000)	

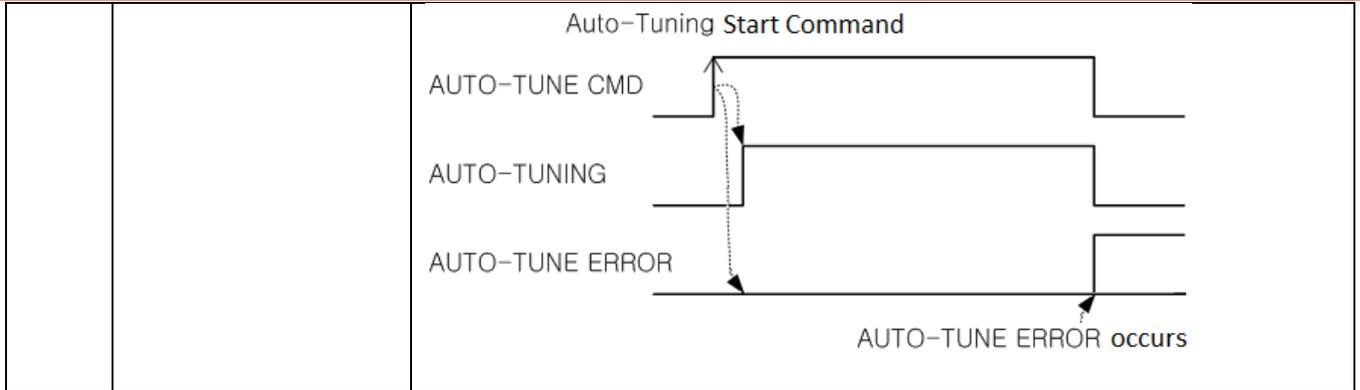
8	Error in Filter Coefficient (α)	When a value is out of the range (0 – 100)	For each loop
9	Error in Manipulated Value Low Limit (MVLL)	When a value is out of the range (0 – 16,000)	
10	Error in Manipulated Value High Limit (MVHL)	When a value is out of the range (0 – 16,000)	
11	Error in Variation Rate Limit of Manipulated Value (Δ MVL)	When a value is out of the range (0 – 16,000)	
12	Error in MV Apply	When a value is neither 0 nor 1	
13	Error in SV Transition Step	When a value is out of the range (0 – 1,000)	



c) PID Flags

Bit	Flag Item	Flag Description
0	PID-INIT OK	Indicates that the initialization of the loop is normally completed by the PIDINIT command. This flag begins the PLC operation and starts the initialization after the sampling time. (0 = error, 1 = normal)
1	PID CALC	This flag is used internally to determine the PID operation interval.
2	CALC DELAY	PID operation is conducted based on the number of loops executed per scan. If the number of loops that should be operated simultaneously exceeds this criterion, the exceeded loop operation is delayed and postponed to the next scan. When this happens, the CALC DELAY flag is set. If this flag is set too often, adjust the number of '1 SCAN Loop' or 'Sampling Interval.'
3	ON/OFF STATUS	This flag is used internally to distinguish between ON and OFF output stage when the 'ON/OFF control function' is used. This flag is set in the ON output stage. However, do not use this function directly in the actual control output. The accurate control of the control output is possible when the MV output of the PIDCAL command is used. It is considered ON output when the MV output value is equal to the value specified in Manipulated Value High

		Limit (MVHL). It is considered OFF output when the MV output value is equal to the value specified in Manipulated Value Low Limit (MVLL).
4	WAIT LOOP INIT	Until the first PIDCAL command is executed, it is in the SET state. This flag is reset when the first PID operation (PIDCAL command) is performed based on the sampling cycle.
5	AUTO TUNE CMD	<p>This flag is set when starting the Auto Tuning function. The target value (SV) must be specified before this flag is set. When the PV value reaches 63% of the target value (SV), PID gains are automatically calculated and written to the relevant areas and thus resets this flag. If the constant calculation fails, the AUTO-TUNE ERROR flag is set and the control is performed using the previous constant.</p> <p>*When shutting down the Auto Tune Cmd, please reset Bit 5, 6 at the same time.</p> <p>Precautions for Auto Tuning The appropriate target value must be specified for direct and reverse operation. For the direct action, a value higher than the PV is entered. For the reverse action, a value lower than the PV is entered. If the deviation between the PV and SV is small, an error may occur or the Auto Tuning does not work properly.</p>
6	AUTO TUNING	<p>This flag is automatically set when Auto Tuning is in progress. When Auto-Tuning is shut down, this flag is reset. Depending on the tuning outcome, AUTO-TUNE ERROR flag can be set (Auto Tuning fails) or the calculated constants are automatically recorded in the relevant areas (Auto Tuning succeeds). If a user wants to stop the operation of the Auto Tuning, which is in progress, the user must reset the bit for AUTO-TUNING and UTO-TUNING.</p>
15	AUTO TUNE ERROR	This flag is set when the constant operation by the Auto Tuning fails. If an error occurs, AUTO-TUNE CMD and AUTO-TUNING bits are reset.



1.4.4 PIDCAL Instruction

The PIDCAL command performs PID calculation based on the SV and PV values and stores the results in the MV area of the word device. Refer to the table below for operation data location.

Offset	Item	Description and Range	Remark
0	Set Value (SV)	0–16,000	
1	Process Value (PV)	0–16,000	
2	Manipulated Value (MV)	0–16,000	
3	PV after filtering (PVnf)	0–16,000	
4	Manual MV (MV _{MAN})	0–16,000	
5	Function Selection Flag	Bit #0 : Automatic (0) / Manual (1) Bit #1 : Enable 'Self-Learning' Bit #2 : Kp x 100 Bit #15 : Request to delete the learned data	For each loop
6	Reserved (Not available to users)		
7			
8			
•			
10		Status	
•			
19			

a) Setting data for each loop

Offset	Item	Description
0	Set Value (SV)	Enter the desired target value that a user wants to control.
1	Process Value (PV)	It is the current value of the controlled unit read from the A/D card. A user either has to update the PV value regularly by reading the value from the A/D card using the "FROM command" or update other measurement values using the "MOV command."
2	Manipulated Value (MV)	It is the result value of the PID operation that needs to be outputted to the control unit by using the D/A card or I/O card.
3	PV after filtering (PVnf)	The filtering is used to prevent instantaneous deviation caused by the PV noise. The filtered PV value is then stored in the PVnf area. When the filter coefficient (α) is 0, the filtering is not applied to the PV. This area can only be monitored.
4	Manual MV (MV _{MAN})	When in manual mode, the value set as the 'manual manipulated value' is outputted as the Manipulated Variable (MV).
5	Function Selection Flag	<p>Automatic / Manual Mode Selection A user gets to select whether to control the controlled unit automatically by using the PID function or to output the value specified in the manual manipulated variable to the controlled unit.</p> <p>Enable Self-Learning Function It is the feature newly added for PID_2 version. The self-learning function allows a user to take advantage of its ability to calculate the most suitable MV output level for the SV based on the result of the previously executed PID control and the control operation on the basis of the existing PID control algorithm. By using this feature, the user can obtain a faster system response.</p> <p>Kp x 100 This function is newly added for PID_2 version. Proportional gain (Kp) can maximize its setting range (1 – 65535) up to 100 times by using this feature. While this flag is set, the range value of Kp can be expanded from 1 to a maximum of 6,553,500. If this flag was to be forcefully reset, it is necessary to reconfigure the [Kp x 100] as the Kp constant in order to obtain the same control effect.</p> <p>Delete Self-Learning data It is the function newly added for PID_2 version. When this flag gets set, the PLC is initialized to delete all of the self-learning data that has already been collected and to build a new learning data. Because the self-learning data is composed of files that are internal-use- only for PLC CPU, the user cannot use this for reference.</p>
10	Status	Displays the status of the relevant loop. *Refer to the table below.

b) Status descriptions for each loop

Bit	Function	Description
0 – 3		Reserved
4	Applying self-learning data	Status when the control is performed by using the collected self-learning data
5	Collecting self-learning data	This status gets activated when the self-learning data is being collected (When the value of SV is changed, the function of collecting the self-learning data gets activated)
6	Stability	Status in which the control system is being stable (PV and SV are the same, and MV output is fixed with no change)
7	Self-learning retention	Status in which the self-learning data is stored in the memory
8	PV inertia amount data retention	This status is set when a non-zero value is stored in the PV inertia amount memory
9 – 15		Reserved

NOTES

1. Prior to the PIDCAL command operation, the PIDINIT command must be operated first.
2. The PID instructions (PIDINIT, PIDCAL) must be used only once in the entire CIMON project.
3. For the use of multiple loops, the PIDINIT area and the PIDCAL area should not be overlapped.
4. Because the PIDINIT area and some areas of the PIDCAL are used for PID operation, if they were set as latch area, the PID function may not work properly.
5. An error occurs when Auto Tuning is running in manual mode. The Auto Tuning function should only be running in automatic mode.

To configure the starting addresses for PIDINIT and PIDCAL

Please make sure that [PIDINIT starting address] and [PIDCAL starting address] do not overlap and that the following calculation is satisfied.

$$[\text{PIDCAL starting address}] \geq [\text{PIDINIT starting address}] + [\text{Total number of Loop}] \times 20 + 2$$

OR

$$[\text{PIDINIT starting address}] \geq [\text{PIDCAL starting address}] + [\text{Total number of Loop}] \times 20$$

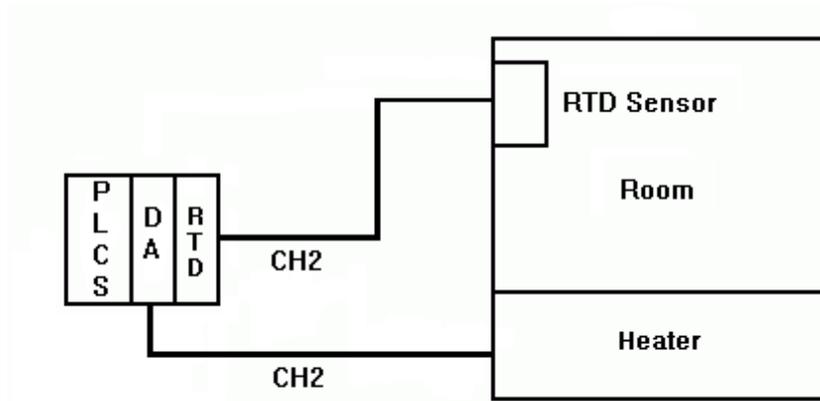
Ex) If the total number of Loops is 10 / “PIDINIT starting address” is 100,

$$< 10 \text{ Loop} \times 20 \text{ Word} + 2 \text{ (common area)} = 202 \text{ --- } > 100 + 202 = 302 >$$

“PIDCAL” starting address should be set as 302 or above.

1.4.5 PID Program Example

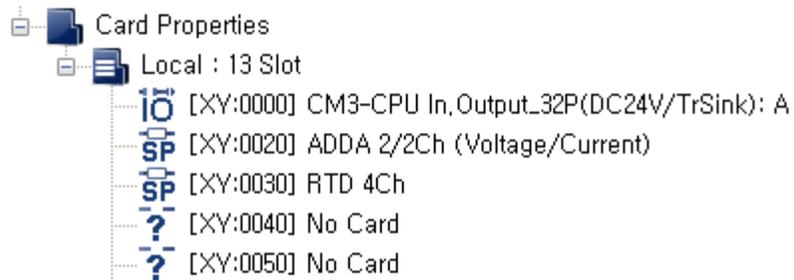
This sample program takes measurement of the temperature inside a room using the RTD sensor and controls 4 – 20 mA output to keep the room temperature as 20 °C. The temperature is measured using the channel 2 of the RTD module, and the output of the heater is controlled using the channel 2 of the DA card.



a) Module Additions

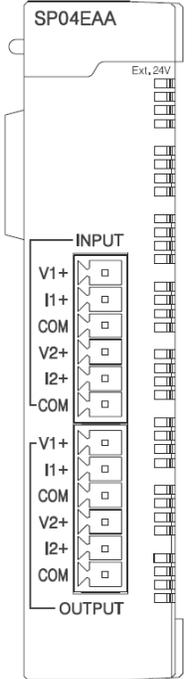
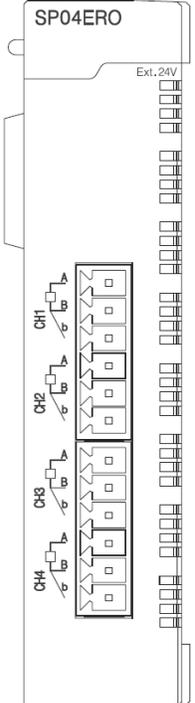
The slot number '0' is PLC-S (CM3-SP32MDT).

The slot number '1' is CM3-SP04EAA while the slot number '2' is CM3-SP04ERO.



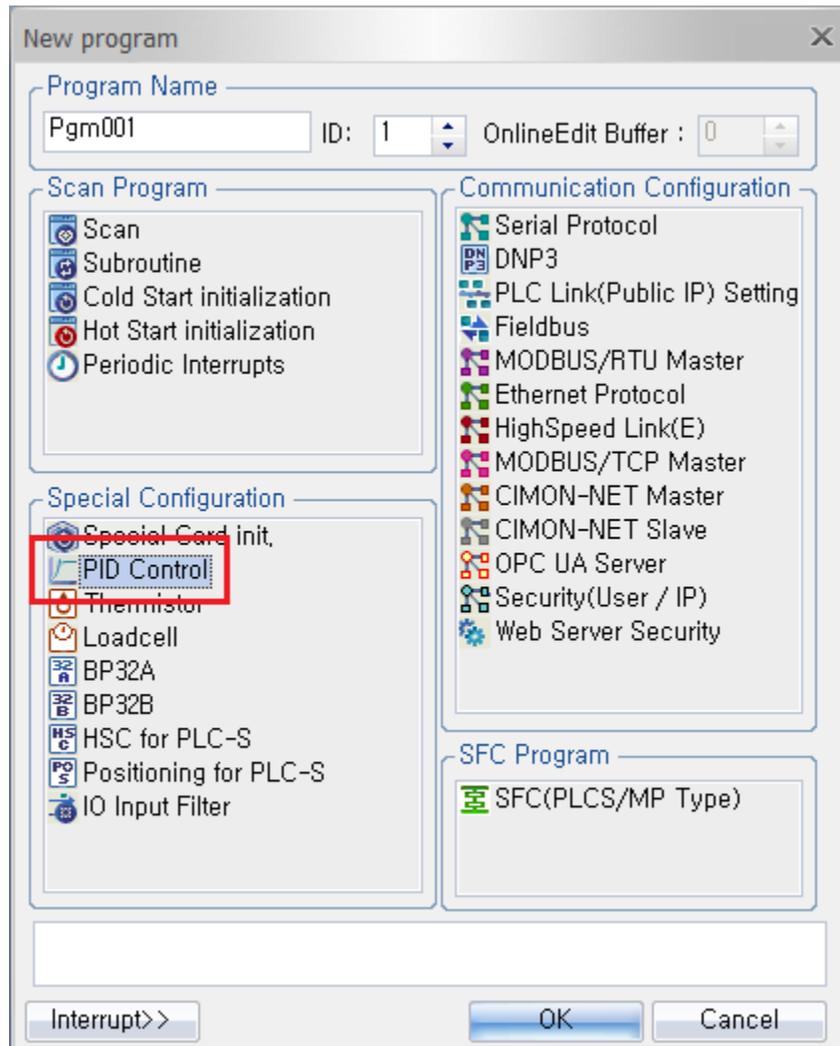
b) Descriptions for Additional Modules

As the above PID test is processed, the following modules are used in addition to PLC-S.

Name	Description	Diagram
<p>SP04EAA (D/A)</p>	<ul style="list-style-type: none"> ▪ CM3-SP04EAA can perform two channels of each A/D conversion and D/A conversion respectively. ▪ Input signals entered as 0~20mA, 4~20mA, 0~5V, -10~10V and 0~10V are converted to digital values from 0~64000 or -32000~32000. ▪ The digital values of 0~64000 or -32000~32000 are outputted as the signal in the range of 0~20mA, 4~20mA, 0~5V, -10~10V and 0~10V. ▪ The internal digital filtering is processed to obtain a precise measurement value. ▪ The averaging process of time and frequency can be applied. The resolution of the digital value can be selected as 1/32000 to obtain a high resolution digital value. 	 <p>The diagram shows the SP04EAA module with a terminal block on the right. The terminal block is divided into two main sections: 'INPUT' and 'OUTPUT'. The 'INPUT' section has two channels, each with terminals for V1+, I1+, COM, V2+, I2+, and COM. The 'OUTPUT' section also has two channels with terminals for V1+, I1+, COM, V2+, I2+, and COM. An 'Ext. 24V' terminal is located at the top right of the terminal block.</p>
<p>SP04ERO (RTD)</p>	<ul style="list-style-type: none"> ▪ Temperature converted by platinum RTD (Pt100 or JPt100) is displayed in Celsius and Fahrenheit (°C, °F). The temperature value can be processed as a digital value to one decimal place. ▪ The converted temperature data is then converted to signed 14 bit binary data and is processed as a digital value from -192~16191. The temperature input (-200°C~600°C or -50°C~160°C) is converted to digital output of 0~16000 (-8000~8000). ▪ Each channel can detect the exceeded measuring range and disconnection of RTD and cables. Four sensors in addition to Pt100 can be used to connect up to 4 points with one module. ▪ EXP LED lights up in normal status and blinks every 0.3 seconds in case of an error. Power LED blinks every 0.5 seconds in case of a communication error with CPU module. 	 <p>The diagram shows the SP04ERO module with a terminal block on the right. The terminal block is divided into four channels, labeled CH1, CH2, CH3, and CH4. Each channel has terminals for RTD input. An 'Ext. 24V' terminal is located at the top right of the terminal block.</p>

c) Registering the PID Special Program

PID Special Program must be registered only once for a CIMON project, and it can have 32 Loops within one program.



d) Configuring the PID Special Program

The screenshot shows the 'Pgm001.SRC' window with the following configuration options:

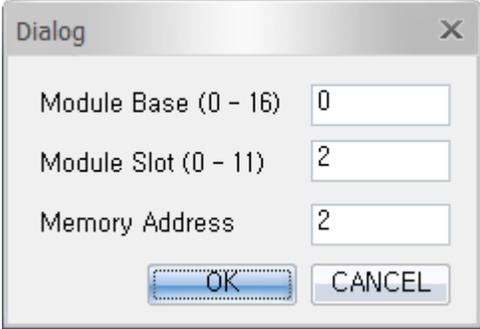
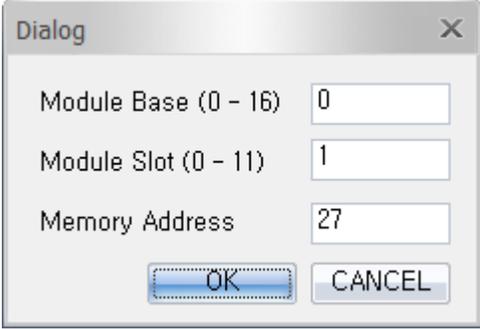
- Buttons: Save, Monitor, Auto Tune, New Dialog, Current Lp. 1, Help
- Current Value -> Set Value: Current Loop, Entire Loops, LD Convert, Trend, Description
- No. of Loop: Total 1, Loops/Scan 1
- Start Data of PID Control: PIDINIT D 0, PIDCAL D 100

Index	Device	Set Value	Current Value
Path Calc(0:Forward 1:Reverse)	D00002	Forward	(e)
Sampling Time(0.01 - 60 sec)	D00003	1.00	(f)
Kp(0 - 65535)	D00004	30000	(g)
Ki(0.0 - 3000 sec)	D00005	20.0	(h)
Kd(0.00 - 300 sec)	D00006	0.00	(i)
Filter(0 - 1.00)	D00007	0.20	(j)
MV Low Limit(0 - 16000)	D00008	0	(k)
MV High Limit(0 - 16000)	D00009	16000	(l)
MV Change Rate Limit(0 - 16000)	D00010	16000	(m)
MV Auto-Apply(0:Disabled 1:Enabled)	D00011	Disable	(n)
SV Ramp(0 - 1000 0:Disabled)	D00012	0	(o)
On/Off Time(0.00 - 60.00)	D00014	0.00	(p)
SV(Set Value)	D00100	200	(q)
PV(Process Value)	D00101	FROM H0003 2 I	(r)
MV(Manipulation Value)	D00102	TO H0001 27 D0	(s)
PVnt(After Filter)	D00103		(t)
MVman(Mv Manual)	D00104	0	(u)
(0:Auto 1:Manual)	D00105	Auto	(v)
Self Learning Enable(1) / Disable(0)	D00105	Enable	(w)
Kp x 100 (1)	D00105	-	(x)
PID Error Code	D00015	0	(y)
PID Status Code	D00016		(z)

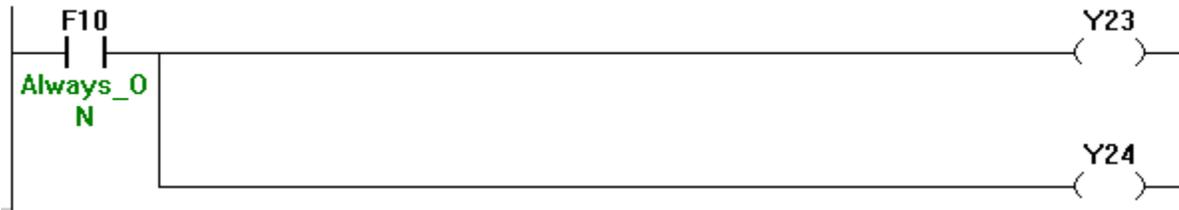
Control Status Indicators:

- PID Loop Initialized
- Auto-tune Processing
- ON/OFF Ctrl Output
- Self Learn Data Ready
- Stable Status
- PID - 2 Control

Number	Item	Description
(a)	Number of Total Loops	Because the system to be controlled is one, select '1.'
(b)	Number of Loops Per Scan	Since the total number of loops is one, select '1' also.
(c)	PIDINIT Start Address	Enter the start address in which PIDINIT setting values are stored. By changing the start address, the device of the setting items get automatically changed.
(d)	PIDCAL Start Address	Enter the start address in which PIDCAL setting values are stored. By changing the start address, the device of the setting items get automatically changed.
(e)	Operation Type	When the manipulated variable increases, so does the measurement value. Therefore, select the forward operation.
(f)	Sampling Time (Ts)	If it is a system where measurement values are changed sensitively for the change of the manipulated variable, the sampling interval should be set short. For a general HVAC system in which the measurement values are insensitive to changes, the sampling interval can be set about 1 second.
(g)	Proportional Gain (K _p)	Enter the proportional gain value of the system. If the hunting of the value is severe, reduce the proportional value. In this test, the value is set to 30,000.
(h)	Integral Gain (K _i)	Enter the integral gain value of the system. If the integral gain value is large, hunting gets reduced but the time to stabilize can be prolonged. In this test, the value is set to 20.
(i)	Derivative Gain (K _d)	Enter the derivative gain value of the system. This value is used when there is severe disturbance or a system with a quick reaction. In this test, it is assumed that there is no serious impact of disturbance. Therefore, this value is set as '0 (Not Use).'
(j)	Filter Coefficient (α)	This value configures to what extent the filtering is going to be applied on the measurement values. The closer the value gets to 0, the less the filtering effect is.
(k)	MV Low Limit	Enter the lowest limit of the manipulated variable. The value is set as 0 in this test.
(l)	MV High Limit	Enter the highest limit of the manipulated variable. Because it is a system in which the manipulated variable value can be outputted more than a specific value, the MV High Limit is set as 16,000 in this test.
(m)	MV Change Rate Limit	Enter the manipulated variable variation rate limit. This value is used to prevent the change of the manipulated variable from occurring suddenly. Since this test requires no specific need to limit on the change of the manipulated variable, the value is set as 16,000.
(n)	MV Auto Apply	This value is used when a user wants to keep the MV value when switching from automatic to manual mode. In this test, the value is set as '0 (Disabled).'
(o)	SV Transition Step	This value is used to control the time taken to reach the set value. In this test, the value is set as '0 (Disabled).'

(p)	On/Off Time	This value is used to control on/off time using the MV value. If a value more than 1 is entered for operation, the MV Low Limit and the MV High Limit will be outputted in turn as the manipulated variable. Because this test does not involve On/Off control, this value is set as '0 (Disabled).'
(q)	SV(Set Value)	Enter the target value that you want to control. (i.e.) '200' is entered.
(r)	PV(Process Value)	<p>Enter the process value of the control unit. You can configure the ERO module and enter the appropriate values as shown in the below dialog image in order to read the data directly. The memory address is set as '2' in order to detect the temperature value of the channel 2 of the ERO module by reading the value from the current directly. As mentioned above, the Module Slot is set as '2' (ERO module being in the second slot) and the Module Base is set as '0' since there is no expansion base.</p> 
(s)	MV(Manipulation Value)	<p>The manipulated variable is the calculated result of the PID operation. You can either get this value by using the MOV command from the external scan program or configuring the DA card directly in order to output the MV value. Enter the value as shown below to use the DA card and to output the MV value directly in this test.</p>  <p>The channel 2's digital output value (DA) of the EAA module is configured as Memory Address '27' (Please check the buffer memory of CM3-SP04EAA in the HELP file for the detailed information) in order to adjust the temperature value through the current. Also as mentioned above, the Module Slot is set as '1' (EAA module being in the first slot) and the Module Base is set as '0' since there is no expansion base.</p>
(t)	Pvnt(After Filter)	Only monitoring is possible.
(u)	MVman(MV Manual)	Use this value when the manipulated variable value is set to manual mode.
(v)	Auto / Manual Selection	This specifies the initial operating mode. In this test, it is 'Auto Mode.'

e) Writing the Scan Program



This scan program enables channel 1 and channel 2 outputs of the EAA module to operate directly.

Refer to the table below for the input and output signals of the PLC-S (CM3)-SP04EAA-CPU.

Signal direction (CPU ← A/D, D/A)		Signal direction (CPU → A/D, D/A)	
Input	Signal Name	Output	Signal Name
X20	A/D module Ready	Y20	
X21		Y21	
X22	Operating Condition Save Complete	Y22	Operating Condition Save Command
X23	CH.1 High Level Alarm	Y23	DA CH.1 Output Enable
X24	CH.2 High Level Alarm	Y24	DA CH.2 Output Enable
X25	CH.3 High Level Alarm	Y25	
X26	CH.4 High Level Alarm	Y26	
X27		Y27	
X28		Y28	
X29		Y29	
X2A		Y2A	
X2B		Y2B	
X2C		Y2C	
X2D		Y2D	
X2E		Y2E	
X2F	D/A Module Error flag	Y2F	Error Clear Command

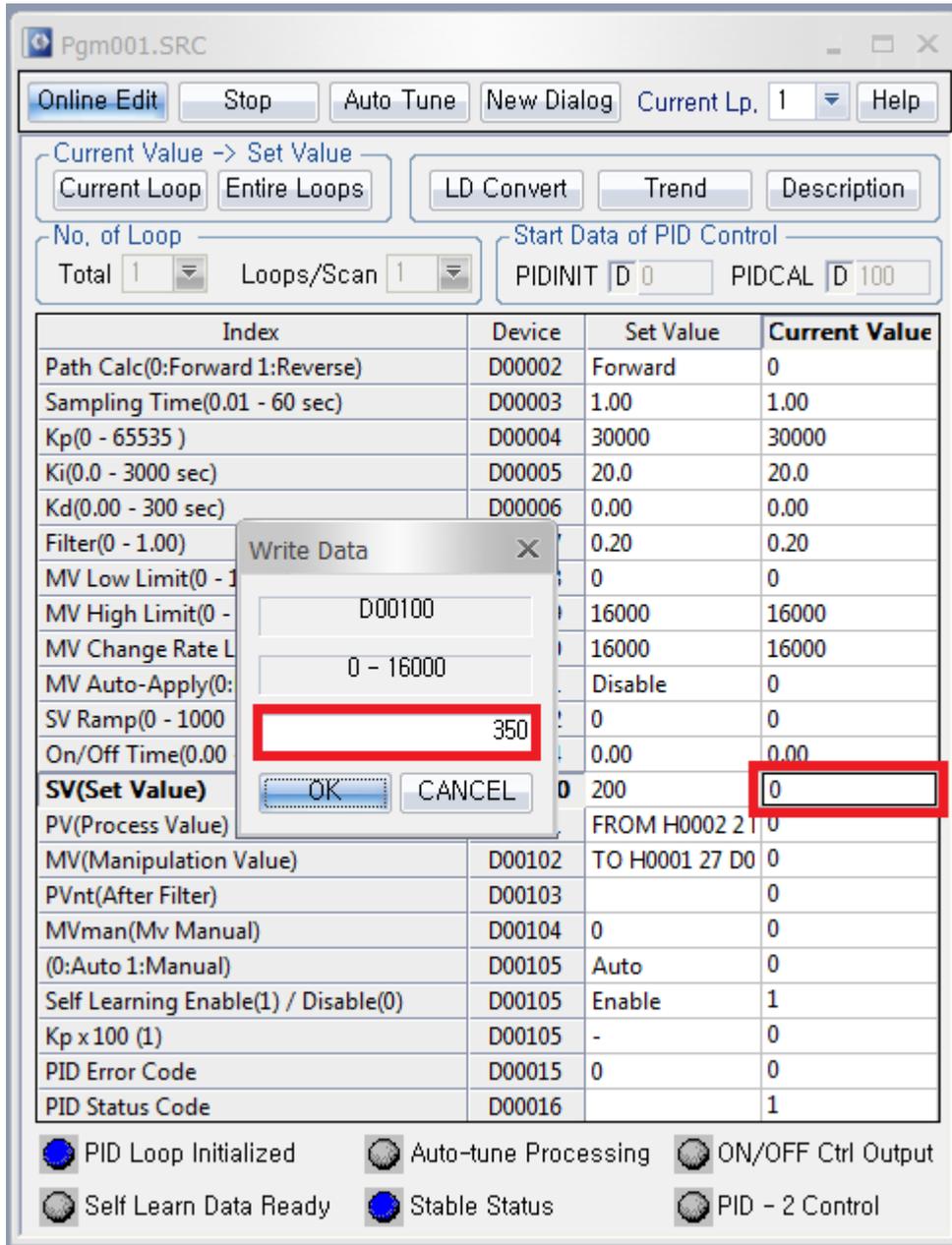
f) Download and Control

The screenshot shows the 'Pgm001.SRC' window with various control options and a parameter table. The 'PID Loop Initialized' status indicator is highlighted with a red box.

Index	Device	Set Value	Current Value
Path Calc(0:Forward 1:Reverse)	D00002	Forward	0
Sampling Time(0.01 - 60 sec)	D00003	1.00	1.00
Kp(0 - 65535)	D00004	30000	30000
Ki(0.0 - 3000 sec)	D00005	20.0	20.0
Kd(0.00 - 300 sec)	D00006	0.00	0.00
Filter(0 - 1.00)	D00007	0.20	0.20
MV Low Limit(0 - 16000)	D00008	0	0
MV High Limit(0 - 16000)	D00009	16000	16000
MV Change Rate Limit(0 - 16000)	D00010	16000	16000
MV Auto-Apply(0:Disabled 1:Enabled)	D00011	Disable	0
SV Ramp(0 - 1000 0:Disabled)	D00012	0	0
On/Off Time(0.00 - 60.00)	D00014	0.00	0.00
SV(Set Value)	D00100	200	200
PV(Process Value)	D00101	FROM H0002 2 I	0
MV(Manipulation Value)	D00102	TO H0001 27 D0	16000
PVnt(After Filter)	D00103		0
MVman(Mv Manual)	D00104	0	0
(0:Auto 1:Manual)	D00105	Auto	0
Self Learning Enable(1) / Disable(0)	D00105	Enable	1
Kp x 100 (1)	D00105	-	0
PID Error Code	D00015	0	0
PID Status Code	D00016		1

PID Loop Initialized
 Auto-tune Processing
 ON/OFF Ctrl Output
 Self Learn Data Ready
 Stable Status
 PID - 2 Control

The fact that the blue LED of the “PID Loop Initialized” is turned ON signifies that the program is operating normally.



Enter the number '350' for the Set Value (SV).

The number '350' signifies 35.0°C. When you want to have the value of 45.5°C, enter the number '455.'

g) Using Auto Tune Function

The screenshot shows the 'Pgm001.SRC' window with the 'Auto Tune' button highlighted. Below it, the 'Trend' button is also highlighted. The main window contains a table of PID parameters and a status bar at the bottom.

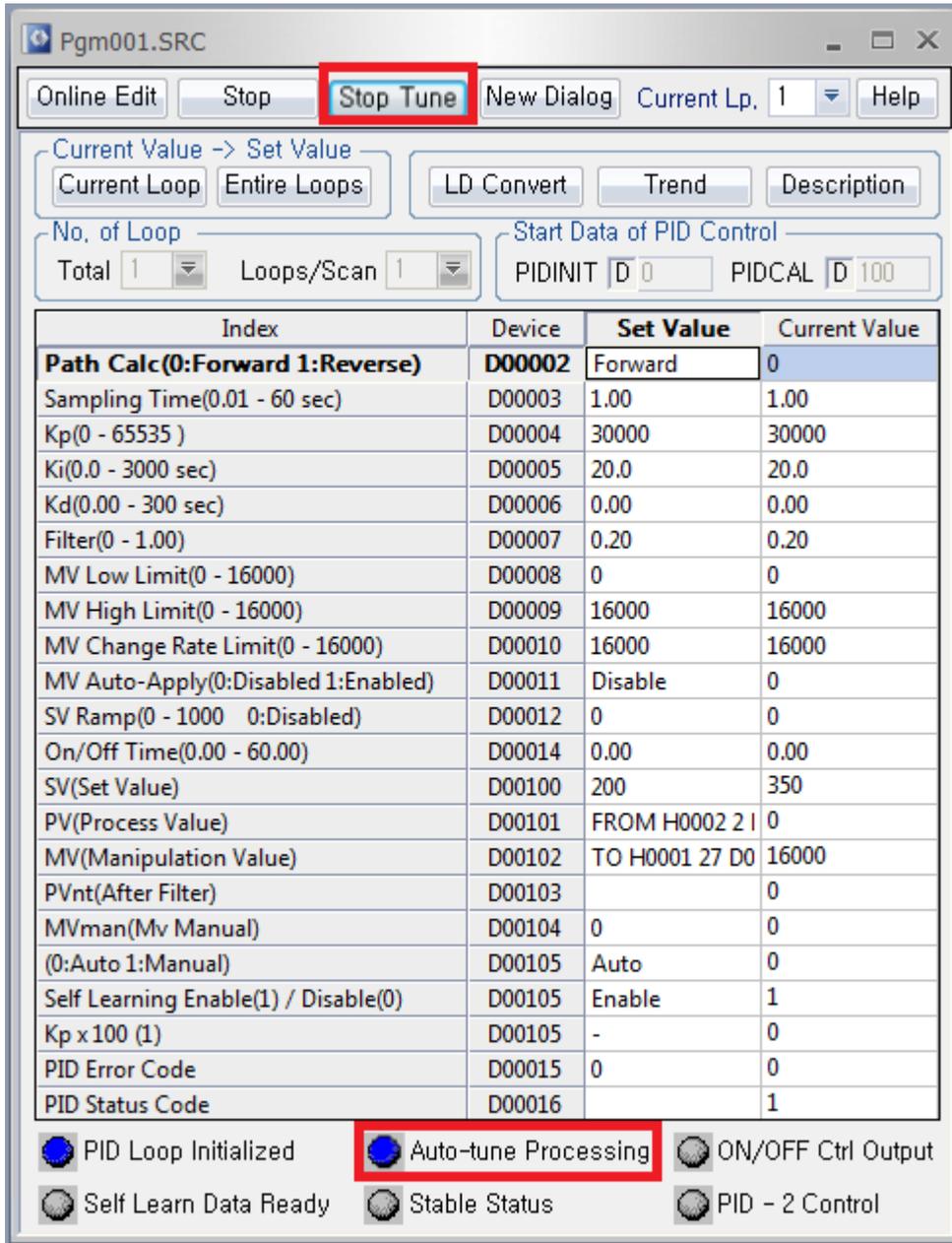
Index	Device	Set Value	Current Value
Path Calc(0:Forward 1:Reverse)			
Sampling Time(0.01 - 60 sec)	D00003	1.00	1.00
Kp(0 - 65535)	D00004	30000	30000
Ki(0.0 - 3000 sec)	D00005	20.0	20.0
Kd(0.00 - 300 sec)	D00006	0.00	0.00
Filter(0 - 1.00)	D00007	0.20	0.20
MV Low Limit(0		0	0
MV High Limit(16000	16000
MV Change Rat		16000	16000
MV Auto-Apply		Disable	0
SV Ramp(0 - 10		0	0
On/Off Time(0.		0.00	0.00
SV(Set Value)	D00100	350	350
PV(Process Value)	D00101	FROM H0002 2	0
MV(Manipulation Value)	D00102	TO H0001 27 D0	16000
PVnt(After Filter)	D00103		0
MVman(Mv Manual)	D00104	0	0
(0:Auto 1:Manual)	D00105	Auto	0
Self Learning Enable(1) / Disable(0)	D00105	Enable	1
Kp x 100 (1)	D00105	-	0
PID Error Code	D00015	0	0
PID Status Code	D00016		1

Auto Tuning dialog box: Set Value: 350

Status indicators: PID Loop Initialized, Auto-tune Processing, ON/OFF Ctrl Output, Self Learn Data Ready, Stable Status, PID - 2 Control

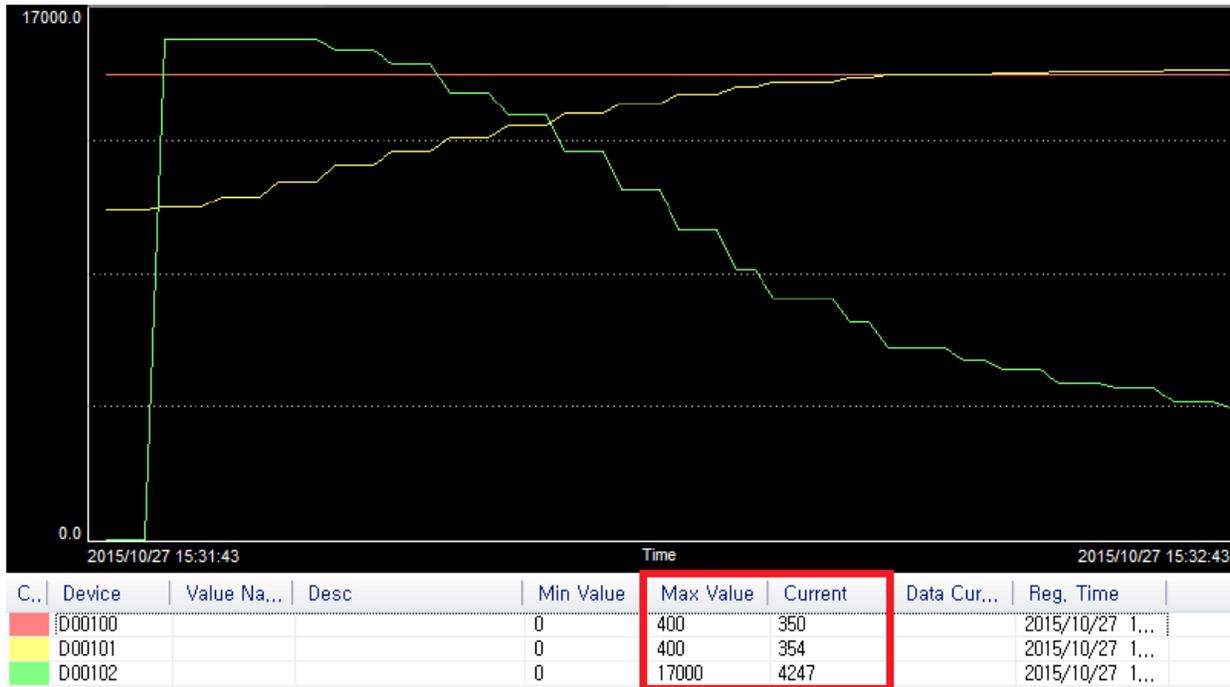
The temperature control is possible by using the Auto Tune function. Depending on the response capability of the system, tuning operations can often be time consuming.

Before the tuning operations, be sure to check whether the flag status of the loop is normal. When the tuning operations are completed, the current value of the proportional, integral, and derivative gain of the loop gets modified.



If you want to cancel the tuning operation, click the button “Stop Tune.” When the Auto Tune function is operating normally, the blue LED of the “Auto-tune Processing” gets turned ON. The target value should be set in a way that is in line with the operation type. In other words, it should be forward direction when the target value is greater than the current PV. It should be reverse direction if the target value is smaller than the current PV. The tuning operation will be automatically completed when the process value reaches about 68 percent of the error between the current PV and the target value.

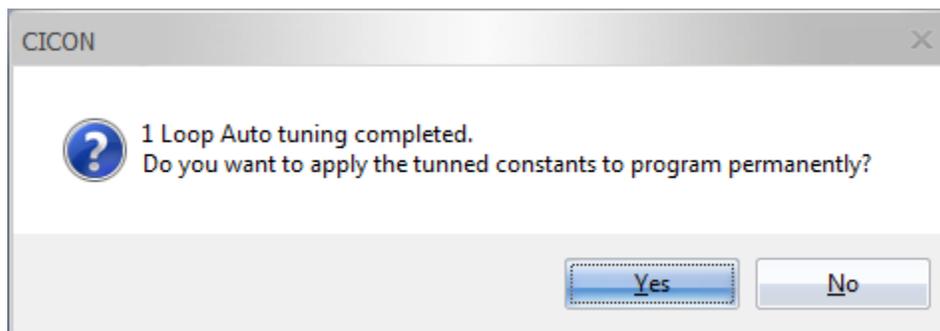
h) Verifying results with a trend graph



Through the Trend function, it is possible to check the status of the SV (Set Value), PV (Process Value), and MV (Manipulation Value).

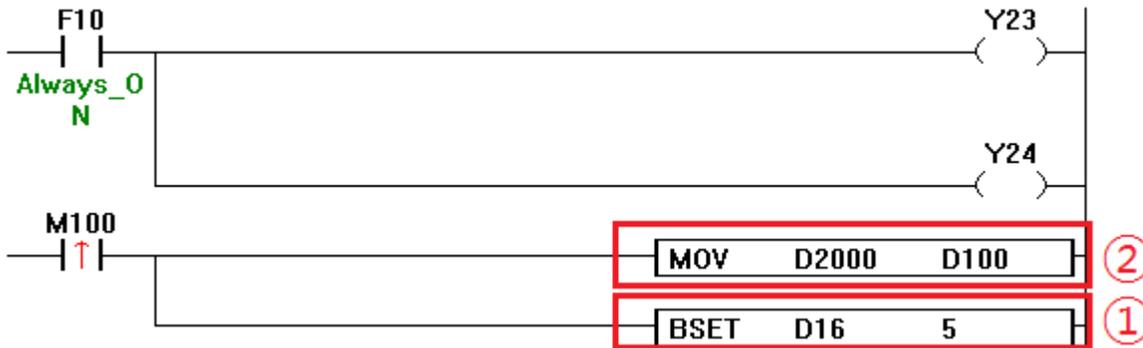
In the image above, the red line signifies the SV while yellow line signifies the PV and the green line signifies the MV. In order to recognize the amount of variation with ease, the max value of the SV and PV is configured as 400. The max value of the MV is set as 17,000.

After the setting value is set as 400, the current value is getting closer to the set value as the temperature changes according to the variation in the amount of MV. Once the PID program is fully stabilized, the current value and the set value almost remain the same without the difference between each other. When Auto Tuning is completed, you will get a message as shown below.



1.4.6 Executing AUTO TUNING in the Sequence Program

a) Writing a Scan Program



When writing a scan program, the conditional equation **D2000 > D100 (direct action)** must be satisfied.

1. BSET D16 5 activates the 5th bit of the device D16.

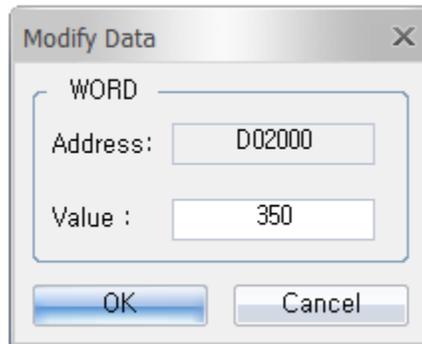
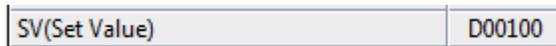
What information is contained in the 5th bit of D16?

The 5th bit is a flag used to start the Auto Tuning as the AUTO-TUNE CMD bit.

(Refer to 1.4.3 PIDINIT Instruction – Setting Data for Each Loop – Flags)

2. MOV D2000 D100 transfers the value of D2000 to D100.

Why use D100? SV (Set Value) is written on the device D100.



When you enter the desired value in D2000, the value of 350 will be written in the SV (D100).

D2000 has been randomly assigned in this test and can be designated by the user as desired.

However, it should be configured to avoid overlapping with other PIDCAL, PIDINIT addresses.