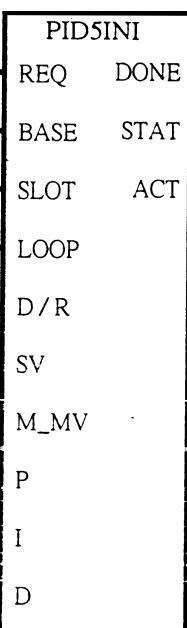


PID5INI

G3F-PIDA module initialization	Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●			

Function block	Description
 <p>[] : Indicate ARRAY variable and number in the parenthesis is the element number.</p>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-PIDA module installation base location number SLOT : Slot location number of G3F-PIDA module installation base <p>Note 1) LOOP : Loop assignment to read handling value. Assign '1' of respective element value for loop assignment.</p> <p>Note 1) D/R : Normal/Reverse operation selection assignment for the operation loop. ('0': Normal, '1': Reverse)</p> <p>Note 1) SV : Control target value input for operation loop (Input value range: 0~16000)</p> <p>Note 1) M_MV : Manual handling value for the operation loop (Input value range: 0~16000)</p> <p>Note 1) P : Proportional constant of the operation loop (0.01~100.00%) (Input value range: 0~10000)</p> <p>Note 1) I : Integral constant of the operation loop (0.0~3000.0 sec.) (Input value range: 0~30000) Integral operation disable in case of I=0.</p> <p>Note 1) D : Differential constant of the operation Loop (0.0~3000.0 sec.) (Input value range: 0~30000) Integral operation disable in case of D=0.</p> <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and holds on till next function block execution is requested. However, off if the error occurs during the function block execution. STAT : Error status display during the function block execution. ACT : Channel display that read the conversion value after completing the function block. The element value of the channel will be '1'. <p>Note 1) ACT : Channel display that read the conversion value after completing the function block. The element value of the channel will be '1'.</p>

■ Function

Assign the preset value for each loop to operate G3F-PIDA module.

■ Program example

PID5RD

G3F-PIDA handling value reading(Single type)

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description
	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request BASE : G3F-PIDA module installation base location number SLOT : Slot location number of G3F-PIDA module installation base LOOP : Loop assignment to be operated PV : Current value input for the operation loop control. (Input value range: 0~16000) A/M : Handling value assignment for the operation loop control. ('0': Automatic operation(PID operation) handling value assignment) ('1': Manual(compulsive) handling value assignment) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if the error occurs or the function block execution is not requested. Note¹⁾ STAT : Error status display during the function block execution. Note¹⁾ MV : Handling value data of each loop assigned by operation loop. (Handling value range: 0~16000)

Function

Input the current value(PV) of operating loop control and read the handling value(MV), Also, the handling value can be selected to PID automatic handling value or manual(compulsive) handling value.

Program example

LD	IL
	<pre> CAL PID5RD PID_RD REQ := %I0.0.0 BASE := BASE SLOT := SLOT LOOP := LOOP PV := PV A / M := AM LD PID_RD.DONE ST %Q 0.1.0 LD PID_RD.STAT ST STAT LD PID_RD.MV ST MV </pre>

POS5 AST

G5F-POPA General automatic positioning operation	Product	GM1	GM2	GM3	GM4	GM5
	Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POSS_AST] direction TB R[REQ] --- FB D[DONE] M[MODL] --- FB S[STAT] A[ACT] --- FB end FB --- R FB --- D FB --- M FB --- S FB --- A </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if the positioning complete signal is received from G5F-POPA and off if the instruction is used according to REQ condition.

■ Function

Instruction to start the positioning operation of G5F-POPA. Check the status of G5F-POPA and use the instruction if the positioning is completed, ACT is ON.

■ Program example

LD	IL
<pre> %I0.0.0 +--+ POS5_AST POS5_AST %Q0.1.0 +--+ REQ - MODL DONE - (%) -> STAT - AST_STAT ACT - AST_ACT +--+ </pre>	<pre> CAL POS5_AST POS5_AST REQ := %I0.0.0 MODL := MODL LD POSS_AST.DONE ST %Q 0.1.0 LD POSS_AST.STAT ST AST_STAT LD POSS_AST.ACT ST AST_ACT </pre>

POS5_CRD

G5F-POPA Current status reading

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_CRD] direction TB FBREQ[REQ] --- FBMODL[MODL] FBMODL --- FBSTAT[STAT] FBSTAT --- FBACT[ACT] FBACT --- FBCA[CA] FBCA --- FBcv[CV] FBcv --- FBMCD[MCD] FBMCD --- FBCDN[CDN] FBCDN --- FBSTAT FBSTAT --- FBDONE[DONE] end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution. CA : Current location address display CV : Current operation speed display MCD : Current M Code value display CDN : Current operation Data No. display

Function

Function block to monitor current operation status of G5F-POPA module.

Program example

LD	IL
<pre> LD %I0.0.0 --> POS5_CRD.DONE MODL --> POS5_CRD.MODL POS5_CRD.DONE --> %Q0.1.0 POS5_CRD.STAT --> CAD_STAT POS5_CRD.CA --> CRD_CA POS5_CRD.CV --> CRD_cv POS5_CRD.MCD --> CRD_MCD POS5_CRD.CDN --> CRD_CDN </pre>	<pre> IL CAL POS5_CRD REQ := %I0.0.0 MODL := MODL POS5_CRD.DONE %Q0.1.0 POS5_CRD.STAT CRD_STAT POS5_CRD.CA CRD_CA POS5_CRD.CV CRD_cv POS5_CRD.MCD CRD_MCD POS5_CRD.CDN CRD_CDN </pre>

POS5_EMG

G5F-POPA Emergency stop

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_EMG] direction TB R[REQ] --- FB D[DONE] --- FB M[MODL] --- FB S[STAT] --- FB end R --- E1[] E1 --- D M --- E2[] E2 --- S </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Execute emergency stop during operating G5F-POPA module. Emergency stop is separated with external emergency stop by external signal and internal emergency stop by function block. In case of internal emergency stop, external emergency stop signal must be always B contact(NC: Normal Close) status. This instruction has priority for other function block.

Program example

LD	IL
<pre> graph LR I0[%I0.0.0] --- R1[REQ] --- FB1[POS5_EMG] C1[%Q0.1.0] --- R2[REQ] --- FB2[POS5_EMG] C2[%I0.0.0] --- M1[MODL] --- FB1 C3[%I0.0.0] --- M2[MODL] --- FB2 S1[STAT] --- O[EMG_STAT] </pre>	<pre> CAL POS5_EMG POS5_EMG REQ . = %I0.0.0 MODL : = MODL LD POS5_EMG.DONE ST %Q0.1.0 LD POS5_EMG.STAT ST EMG_STAT </pre>

POS5_FLT

G5F-POPA Floating zero point set

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_FLT] direction TB R[REQ] --- FB M1[MODL] --- FB D[DONE] --- FB S[STAT] --- FB end I1[REQ] --- R I2[MODL] --- M1 O1[DONE] --- D O2[STAT] --- S </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Instruction to use current position as zero compulsively instead of machine zero. Current position will be the value of zero address at the parameter.

Program example

LD	IL
<pre> graph LR I1[%I0.0.0] --- FB[POS5_FLT] FB --- I2[%Q0.1.0] FB --- S[STAT] --- FLS[FLT_STAT] M1[MODL] --- FB </pre>	<pre> CAL POS5_FLT POS5_FLT REQ := %I0.0.0 MODL := MODL POSS_FLT.DONE %Q0.1.0 POSS_FLT.STAT FLT_STAT LD ST LD ST </pre>

POS5_INC

G5F-POPA Inchng

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description		
	Input <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number ROT : Set the rotational direction of inching operation ('0': FWD, '1': BWD) INCH_AMT : Set the pulse amount to output (1~99) Output <ul style="list-style-type: none"> DONE : On if the function block is execute without error and off if ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if the order processing complete signal is received from G5F-POPA and off if the instruction is used according to REQ condition. 		

Function

As the manual operation, output the pulse of the value set by rotational direction and INCH_AMT whenever REQ input signal is toggled.

Program example

LD	IL
	<pre> CAL POSS_INC POSS5_INC REQ := %I0.0.0 MODL := MODL ROT := ROT INCH_AMT := INCH_AMT LD POSS5_INC.DONE ST %Q0.1.0 LD POSS5_INC.STAT ST FLT_STAT LD POSS5_INC.ACT ST INC_ACT </pre>

POS5_JOG

G5F-POPA JOG Operation

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_JOG] direction TB IN_REQ[REQ] --> FB IN_MODL[MODL] --> FB IN_ROT[ROT] --> FB IN_HL[HL] --> FB OUT_DONE[DONE] --> FB OUT_STAT[STAT] --> FB OUT_JOGSTAT[JOG_STAT] --> FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request MODL : G5F-POPA module location number ROT : Set the rotational direction of JOG operation ('0': FWD, '1': BWD) HL : Set the high-speed/low-speed of JOG operation. ('0': Low-speed, '1': High-speed) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till REQ condition is off. STAT : Display the error code generated during the function block execution.

Function

Instruction for G5F-POPA module to execute JOG operation which is used for manual operation and test operation. JOG operation is executed when REQ input is on and stopped when REQ input is off.

Program example

LD	IL
<pre> graph LR I000[%I0.0.0] --> FB[POS5_JOG] MODL --> FB ROT --> FB HL --> FB FB.DONE --> NC(()) NC --> FB </pre>	<pre> CAL POS5_JOG POS5_JOG REQ := %I0.0.0 MODL := MODL ROT := ROT HL := HL LD POS5_JOG.DONE ST %Q0.1.0 LD POS5_JOG.STAT ST JOG_STAT </pre>

POS5_MOF

G5F-POPA M Code Off

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Instruction to switch off M Code On signal during which is set during positioning operation as the M code mode (with, after) in parameter.

Program example

LD	IL
	<pre> CAL POS5_MOF POS5_MOF REQ := %I0.0.0 MODL := MODL LD POS5_MOF.DONE ST %Q0.1.0 LD POS5_MOF.STAT ST MOF_STAT </pre>

POS5_NM

G5F-POPA Next Move

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_NM] direction TB R[REQ] --- FB M1[MODL] --- FB S1[STAT] --- FB A1[ACT] --- FB D1[DONE] --- FB S2[NM_STAT] --- FB A2[NM_ACT] --- FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if the order processing complete signal is received from G5F-POPA and off if the instruction is used according to REQ condition.

Function

Instruction to operate continuously without stop from current operation speed to next operation speed.

Program example

LD	IL
<pre> graph LR I1[%I0.0.0] --- R1[REQ] I2[MODL] --- R2[MODL] R1 --- FB[POS5_NM] R2 --- FB FB --- O1[%Q0.1.0] FB --- NM_STAT[NM_STAT] FB --- NM_ACT[NM_ACT] </pre>	<pre> CAL POS5_NM POS5_NM REQ := %I0.0.0 MODL := MODL LD POS5_NM.DONE %Q 0.1.0 LD POS5_NM.STAT NM_STAT LD POS5_NM.ACT NM_ACT ST ST LD ST LD ST </pre>

POS5_OFF

G5F-POPA Output prohibit release

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR A[REQ] --> B[POS5_OFF] C[MODL] --> B B -- DONE --> D[BOOL] B -- STAT --> E[USINT] </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Instruction to release the Output prohibit after the output is prohibited by emergency stop or upper/lower limit error.

Program example

LD	IL
<pre> graph LR I1[%I0.0.0] --> R1[REQ] R1 --> B1[POS5_OFF] B1 -- MODL --> B2[POS5_OFF] B1 -- DONE --> C1(()) C1 --> R2[REQ] R2 --> B2 B1 -- STAT --> O1[OFF_STAT] </pre>	<pre> CAL POS5_OFF POS5_OFF REQ := %I0.0.0 MODL := MODL LD POS5_OFF.DONE %Q 0.1.0 ST POS5_OFF.STAT OFF_STAT LD ST </pre>

POS5_OR

G5F-POPA Override

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_OR] direction TB IN_REQ[REQ] --> FB IN_MODL[MODL] --> FB IN_OVR[OVR] --> FB OUT_DONE[DONE] --> OUT_Done[OUT_DONE] OUT_STAT[STAT] --> OUT_Stat[OUT_STAT] OUT_ACT[ACT] --> OUT_Act[OUT_ACT] end IN_REQ --- FB IN_MODL --- FB IN_OVR --- FB OUT_DONE --- OUT_Done OUT_STAT --- OUT_Stat OUT_ACT --- OUT_Act </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number OVR : Change the speed from 10 to 150% based on predefined speed and set 1~15 value(Override value x 10% speed change) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G5F-POPA and off if the instruction is used according to REQ condition.

Function

Instruction to change the speed during the operation that can change the speed from 10 to 150% by 10%.

Program example

LD	IL
<pre> graph LR I0["%I0.0.0"] --- FB_REQ[REQ] I0 --- FB_MODL[MODL] I0 --- FB_OVR[OVR] FB_DONE[DONE] --- OR_IN1[OR_IN1] Q0["%Q0.1.0"] --- OR_IN2[OR_IN2] OR_IN1 --- OUT_Q0["%Q0.1.0"] OR_IN2 --- OUT_Q0 </pre>	<pre> CAL POSS5_OR POSS5_OR REQ := %I0.0.0 MODL := MODL OVR := OVR POSS5_OR.DONE %Q 0.1.0 POSS5_OR.STAT OR_STAT POSS5_OR.ACT OR_ACT </pre>

POS5_ORG

G5F-POPA Zero point return	Product	GM1	GM2	GM3	GM4	GM5
Applicable						●

Function block	Description
	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the zero point return complete signal from G5F-POPA and off if the instruction is used according to REQ condition.

Function

Instruction to find the machine zero point as the direction and speed set by parameter. The machine zero point return operation is completed when receives the zero point return complete signal.

Program example

LD	IL
	<pre> CAL POSS_ORG POSS_ORG REQ := %I0.0.0 MODL LD POSS_ORG.DONE %Q0.1.0 ST LD POSS_ORG.STAT ORG_STAT ST LD POSS_ORG.ACT ORG_ACT ST </pre>

POS5_PRE

G5F-POPA Preset

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR A[REQ] --- B[MODL] A --- C[PRESET] B --- D[DONE] C --- E[STAT] C --- F[PRE_STAT] </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number PRESET : Set the data including sign(-16,744,447~+16,744,447) to change current position to certain value. <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Instruction to change current position to certain value within the range of -16,744,447 ~ +16,744,447.

Program example

LD	IL
<pre> graph LR I0[%(I0.0.0)] --- B[POS5_PRE] B --- D[DONE] B --- E[STAT] B --- F[PRE_STAT] C[%(Q0.1.0)] --- B C --- E G[%(Q0.1.0)] --- B G --- E D --- H[PRE_STAT] </pre>	<pre> CAL POSS5_PRE POSS5_PRE REQ := %I0.0.0 MODL := MODL PRESET := PRESET LD POSS5_PRE.DONE ST %Q 0.1.0 LD POSS5_PRE.STAT ST PRE_STAT </pre>

POS5_RES

G5F-POPA Error Reset	Product	GM1	GM2	GM3	GM4	GM5
Applicable						●

Function block	Description
<pre> graph LR subgraph FB [POS5_RES] direction TB R[REQ] --- FB D1[DONE] M[MODL] S[STAT] FB --- D1 FB --- S end R --- B1[BOOL] D1 --- B2[BOOL] M --- U1[USINT] S --- U2[USINT] </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Instruction to reset the error generated at G5F-POPA module. However, the output prohibit status cannot be released. For this, use the POS5_OFF.

Program example

LD	IL
<pre> graph LR I1[%I0.0.0] --- R1[REQ] R1 --- D1[double coil] D1 --- R2[REQ] D1 --- D2[DONE] D2 --- C1[%Q0.1.0] M1[MODL] --- C2[coil] S1[STAT] --- C3[coil] </pre>	<pre> CAL POS5_RES POS5_RES REQ := %I0.0.0 MODL := MODL LD POS5_RES.DONE %Q 0.1.0 ST POS5_RES.STAT PRE_STAT LD POS5_RES.STAT PRE_STAT ST POS5_RES.STAT PRE_STAT </pre>

POS5_RTP

G5F-POPA Return to Position

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_RTP] direction TB REQ[REQ] --- FB MODL1[MODL] --- FB STAT1[STAT] --- FB ACT1[ACT] --- FB DONE[DONE] --- FB STAT2[STAT] --- FB ACT2[ACT] --- FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G5F-POPA and off if the instruction is used according to REQ condition.

■ Function

Instruction to return the position before manual operation start when the position is changed by manual operation.

■ Program example

LD	IL
<pre> LD %I0.0.0 --- REQ MODL --- MODL OUT --- DONE </pre>	<pre> IL CAL POS5_RTP REQ := %I0.0.0 MODL := MODL LD ST LD ST LD ST </pre> <p style="text-align: right;">POSS_RTP.DONE %Q 0.1.0 RTP_STAT RTP_ACT</p>

POS5_SMC

G5F-POPA Next execution data number change

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description	
<pre> graph LR subgraph FB [POS5_SMC] direction TB R[REQ] --> FB M1[MODL] --> FB S1[ST_SET] --> FB D1[DONE] S2[STAT] end R --- FB M1 --- FB S1 --- FB D1 --- FB S2 --- FB </pre>	Input REQ : Function block execution request at rising edge MODL : G5F-POPA module location number ST_SET : Change the operation data no. at next instruction within the range of 0~299.	Output DONE : On if the function block is executed without error and off if hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Instruction to change the operation data no that will be processed by the next instruction within the range of 0~299.

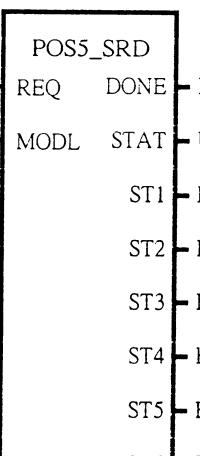
Program example

LD	IL
<pre> graph LR I000[%I0.0.0] --> POS5_SMC[POS5_SMC] POS5_SMC -- REQ --> I000 POS5_SMC -- MODL --> MODL[MODL] POS5_SMC -- ST_SET --> ST_SET[ST_SET] POS5_SMC -- DONE --> Q010[%Q0.1.0] POS5_SMC -- STAT --> SMC_STAT[SMC_STAT] </pre>	<pre> CAL POSS_SMC POSS_SMC REQ := %I0.0.0 MODL := MODL ST_SET := ST_SET LD POSS_SMC.DONE POSS_SMC.DONE ST %Q 0.1.0 LD POSS_SMC,STAT POSS_SMC,STAT ST SET_STAT </pre>

POS5_SRD

G5F-POPA Bit information reading of current operation status

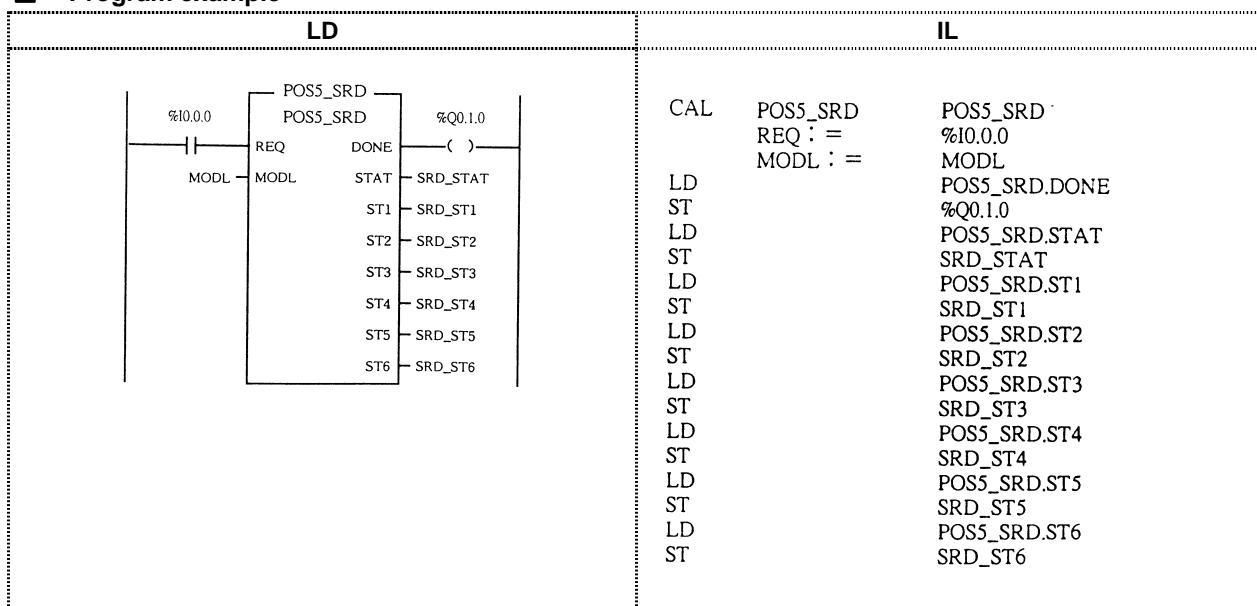
Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block		Description																																																																										
BOOL	REQ	DONE	BOOL	Input REQ : Function block execution request at rising edge MODL : G5F-POPA module location number																																																																								
USINT	MODL	STAT	USINT	Output DONE : On if the function block is executed without error and off if hold on till next function block is executed. STAT : Display the error code generated during the function block execution.																																																																								
	ST1	BOOL[8]																																																																										
	ST2	BOOL[8]																																																																										
	ST3	BOOL[8]																																																																										
	ST4	BOOL[8]																																																																										
	ST5	BOOL[8]																																																																										
	ST6	BOOL[8]																																																																										
		<table border="1"> <thead> <tr> <th></th><th>ST1 content</th><th>ST2 content</th><th>ST3 content</th></tr> </thead> <tbody> <tr> <td>[0]</td><td>Dwell state</td><td>Upper limit</td><td>Not used</td></tr> <tr> <td>[1]</td><td>Decel state</td><td>Lower limit</td><td>FWD/BWD</td></tr> <tr> <td>[2]</td><td>Static state</td><td>Emergency stop</td><td>ZONE#1</td></tr> <tr> <td>[3]</td><td>Accel state</td><td>Pulse output prohibit</td><td>ZONE#2</td></tr> <tr> <td>[4]</td><td>Stop state</td><td>Inching completion</td><td>ZONE#3</td></tr> <tr> <td>[5]</td><td>Zeroing state</td><td>Teaching completion</td><td>Repeat operation completion</td></tr> <tr> <td>[6]</td><td>Positioning state</td><td>JOG low-speed operation</td><td>Positioning start completion</td></tr> <tr> <td>[7]</td><td>Not used</td><td>JOG high-speed operation</td><td>M Code On</td></tr> </tbody> </table> <table border="1"> <thead> <tr> <th></th><th>ST4 content</th><th>ST5 content</th><th>ST6 content</th></tr> </thead> <tbody> <tr> <td>[0]</td><td>Zeroin compensation</td><td>Stop</td><td>Not used</td></tr> <tr> <td>[1]</td><td>Backlash compensation</td><td>Upper limit (H/W)</td><td>Error</td></tr> <tr> <td>[2]</td><td>Next Move processing</td><td>Lower limit (H/W)</td><td>Position passing signal</td></tr> <tr> <td>[3]</td><td>Override processing</td><td>Dog signal (H/W)</td><td>Busy</td></tr> <tr> <td>[4]</td><td>Not used</td><td>Zero point signal (H/W)</td><td>Positioning completion</td></tr> <tr> <td>[5]</td><td>Decel stop and completion</td><td>Not used</td><td>Zero point return completion</td></tr> <tr> <td>[6]</td><td>Speed teaching completion</td><td>Not used</td><td>Zero point not defined</td></tr> <tr> <td>[7]</td><td>Speed change completion</td><td>Emergency stop (H/W)</td><td>Not used</td></tr> </tbody> </table>				ST1 content	ST2 content	ST3 content	[0]	Dwell state	Upper limit	Not used	[1]	Decel state	Lower limit	FWD/BWD	[2]	Static state	Emergency stop	ZONE#1	[3]	Accel state	Pulse output prohibit	ZONE#2	[4]	Stop state	Inching completion	ZONE#3	[5]	Zeroing state	Teaching completion	Repeat operation completion	[6]	Positioning state	JOG low-speed operation	Positioning start completion	[7]	Not used	JOG high-speed operation	M Code On		ST4 content	ST5 content	ST6 content	[0]	Zeroin compensation	Stop	Not used	[1]	Backlash compensation	Upper limit (H/W)	Error	[2]	Next Move processing	Lower limit (H/W)	Position passing signal	[3]	Override processing	Dog signal (H/W)	Busy	[4]	Not used	Zero point signal (H/W)	Positioning completion	[5]	Decel stop and completion	Not used	Zero point return completion	[6]	Speed teaching completion	Not used	Zero point not defined	[7]	Speed change completion	Emergency stop (H/W)	Not used
	ST1 content	ST2 content	ST3 content																																																																									
[0]	Dwell state	Upper limit	Not used																																																																									
[1]	Decel state	Lower limit	FWD/BWD																																																																									
[2]	Static state	Emergency stop	ZONE#1																																																																									
[3]	Accel state	Pulse output prohibit	ZONE#2																																																																									
[4]	Stop state	Inching completion	ZONE#3																																																																									
[5]	Zeroing state	Teaching completion	Repeat operation completion																																																																									
[6]	Positioning state	JOG low-speed operation	Positioning start completion																																																																									
[7]	Not used	JOG high-speed operation	M Code On																																																																									
	ST4 content	ST5 content	ST6 content																																																																									
[0]	Zeroin compensation	Stop	Not used																																																																									
[1]	Backlash compensation	Upper limit (H/W)	Error																																																																									
[2]	Next Move processing	Lower limit (H/W)	Position passing signal																																																																									
[3]	Override processing	Dog signal (H/W)	Busy																																																																									
[4]	Not used	Zero point signal (H/W)	Positioning completion																																																																									
[5]	Decel stop and completion	Not used	Zero point return completion																																																																									
[6]	Speed teaching completion	Not used	Zero point not defined																																																																									
[7]	Speed change completion	Emergency stop (H/W)	Not used																																																																									
[] : Indicate ARRAY variable and number in the parenthesis is the element number.																																																																												

■ Function

Function block to monitor current operation status of G5F-POPA module by bit information.

■ Program example



POS5_TEA

G5F-POPA Position Teaching

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description	
	Input REQ : Function block execution request at rising edge MODL : G5F-POPA module location number ST_SET : Set the operation data no. for position teaching.(0~299) PRESET : Set the position address for position teaching.(-16,744,447~ +16,744,447)	Output DONE : On if the function block is executed without error and off if hold on till next function block is executed. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G5F-POPA and off if the instruction is used according to REQ condition.

Function

Set the position address of certain operation data. The zero point must have defined before.

Program example

LD	IL
	<pre> CAL POSS_TEA REQ := %I0.0.0 MODL := MODL ST_SET := ST_SET PRESET := PRESET LD POSS_TEA.DONE ST %Q0.1.0 LD POSS_TEA.STAT ST TEA_STAT LD POSS_TEA.ACT ST TEA_ACT </pre>

POS5_TMP

Deceleration stop

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_TMP] direction TB IN_REQ[REQ] --> FB IN_MODL[MODL] --> FB IN_ACT[ACT] --> FB OUT_DONE[DONE] --> FB OUT_STAT[STAT] --> FB OUT_TMP_STAT[TMP_STAT] --> FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off when ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G5F-POPA and off if the instruction is used according to REQ condition.

Function

Instruction to stop G5F-POPA during operation. if the function block relating stop is processed by POS5_AST, ACT condition of POS5_AST function block is not on. The machine will be operated again by toggling the function block input REQ of POS5_AST,

Program example

LD	IL
<pre> graph LR I0[%(I0.0.0)] --- FB[POS5_TMP] C1(()) --- FB.DONE </pre>	<pre> CAL POSS_TMP REQ := %(I0.0.0) MODL := MODL LD POSS_TMP.DONE ST %Q0.1.0 LD POSS_TMP.STAT ST TMP_STAT LD POSS_TMP.ACT ST TMP_ACT </pre>

POS5_VCG

Speed change	Product	GM1	GM2	GM3	GM4	GM5
Applicable						●

Function block	Description		
	Input REQ : Function block execution request at rising edge MODL : G5F-POPA module location number VEL_SET : Set current speed to the speed value to be changed.	Output DONE : On if the function block is executed without error and off when ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G5F-POPA and off if the instruction is used according to REQ condition.	

Function

Instruction to change the speed during operation under the static speed only. Available at single operation, repeat operation, JOG and zero point return in high-speed operation.

Program example

LD	IL
	<pre> CAL POS5_VCG POS5_VCG REQ := %I0.0.0 MODL := MODL VER_SET := VER_SET LD POS5_VCG.DONE ST LD POS5_VCG.STAT ST VCG_STAT LD POSS_VCG.ACT ST VCG_ACT </pre>

POS5_VLT

Speed Teaching

Product	GM1	GM2	GM3	GM4	GM5
Applicable					●

Function block	Description
<pre> graph LR subgraph FB [POS5_VLT] direction TB IN_REQ[REQ] --> FB IN_MODL[MODL] --> FB IN_VEL_NO[VEL_NO] --> FB IN_VEL_SET[VEL_SET] --> FB OUT_DONE[DONE] --> OUT_DONE OUT_STAT[STAT] --> OUT_STAT OUT_ACT[ACT] --> OUT_ACT end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge MODL : G5F-POPA module location number VEL_NO : Set the speed data no. for speed teaching.(0~127) VEL_SET : Set the speed value for speed teaching (1~20,000) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off till next function block is executed. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G5F-POPA and off if the instruction is used according to REQ condition.

Function

The speed teaching instruction sets certain speed to the certain value of speed data no.

Program example

LD	IL
<pre> LD %I0.0.0 --> POS5_VLT.DONE MODL --> POS5_VLT.MODL VEL_NO --> POS5_VLT.VER_NO VEL_SET --> POS5_VLT.VER_SET POS5_VLT.DONE --> %Q0.1.0 POS5_VLT.VER_STAT --> VLT_STAT POS5_VLT.ACT --> VLT_ACT </pre>	<pre> IL CAL POS5_VLT POS5_VLT REQ := %I0.0.0 MODL := MODL VER_NO := VEL_NO VER_SET := VEL_SET POS5_VLT.DONE %Q0.1.0 POS5_VLT.VER_STAT VLT_STAT POS5_VLT.ACT VLT_ACT </pre>

POSA_AST

G3F-POAA General automatic positioning operation	Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●			

Function block	Description
	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POPA and G4F-POPA module installation base location number SLOT : Slot location number of G3F-POPA and G4F-POPA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the positioning completion signal from G3F-POAA and off if the instruction is used according to REQ condition.

Function

Instruction to start the positioning operation of G3F-POAA. Check the status of G3F-POAA and use the instruction. If the positioning is completed, ACT is ON.

Program example

LD	IL
	<pre> CAL POSA_AST POSA_AST REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := , AXIS LD POSA_AST.DONE ST %Q 0.1.0 LD POSA_AST.STAT ST AST_STAT LD POSA_AST.ACT ST AST_ACT </pre>

POSA_CRD

G3F-POAA current status reading operation

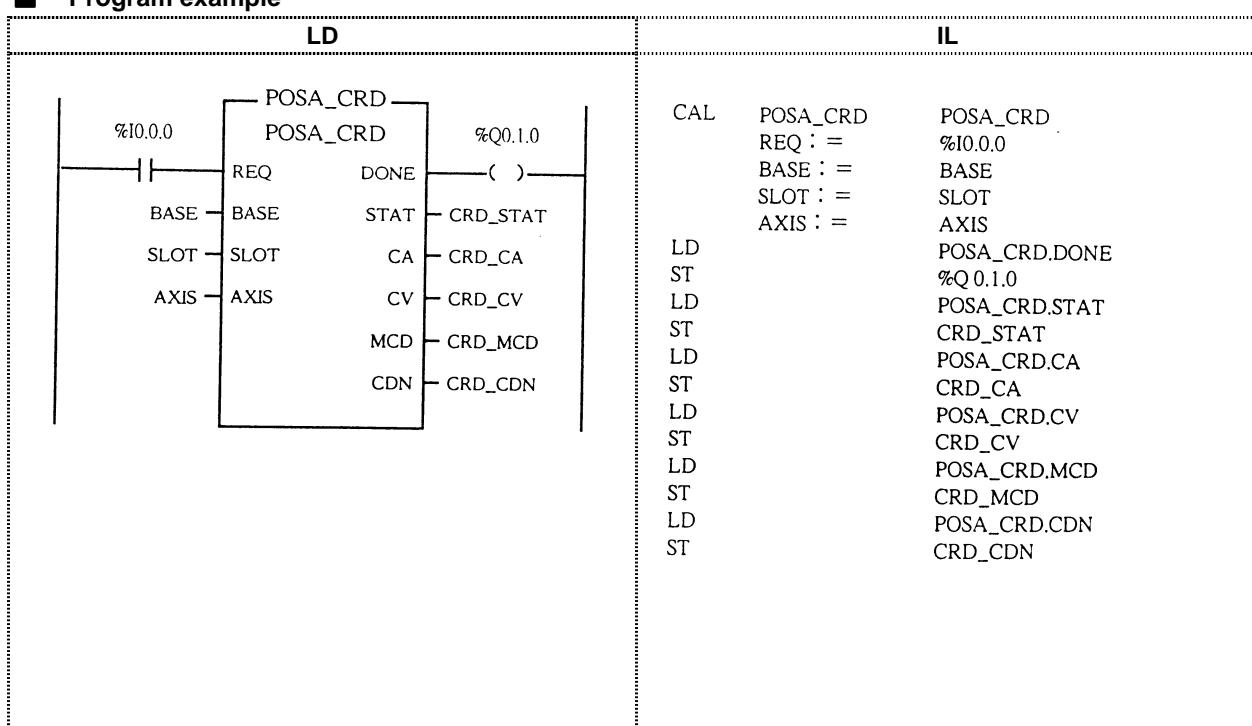
Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description
	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off till next function block is executed. STAT : Display the error code generated during the function block execution. CA : Current position address display CV : Current operation speed display MCD : Current M Code value display CDN : Current operation data No. display

■ Function

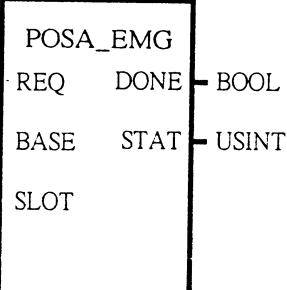
Function block to monitor current operation status of G3F-POAA module.

■ Program example



POSA EMG

G3F-POAA Emergency stop	Product	GM1	GM2	GM3	GM4	GM5
	Applicable	●	●	●		

Function block	Description
 <pre> graph LR subgraph FB [POSA_EMG] direction TB R1[REQ] --- FB B1[BASE] --- FB S1[SLOT] --- FB FB --> D1[DONE] FB --> S2[STAT] end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off till next function block is executed. STAT : Display the error code generated during the function block execution.

■ Function

Execute emergency stop during operating G3F-POAA and module. Emergency stop is separated with external emergency stop by external signal and internal emergency stop by function block. In case of internal emergency stop, external emergency stop signal shall be always B contact(NC: Normal Close) status. This instruction has priority for other function block.

■ Program example

LD	IL
<pre> LD %I0.0.0 POSA_EMG -----+ REQ -----+ -----+ BASE -----+ SLOT -----+ STAT -----+ DONE -----+ ()-----+ EMG_STAT -----+ </pre>	<pre> CAL POSA_EMG POSA_EMG REQ : = %I0.0.0 BASE : = BASE SLOT : = SLOT LD POSA_EMG.DONE ST %Q0.1.0 LD POSA_EMG.STAT ST EMG_STAT </pre>

POSA_FLT

G3F-POAA Floating zero point set

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description
<pre> graph LR subgraph FB [POSA_FLT] direction TB R[REQ] --- FB B1[BASE] --- FB S1[SLOT] --- FB A1[AXIS] --- FB D1[DONE] --- FB S2[STAT] --- FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Instruction to use current position as zero point compulsively instead of machine zero point. Current position will be the value of zero point address at the parameter.

Program example

LD	IL
<pre> graph LR I1[%I0.0.0] --- FB1[POSA_FLT] FB1 --- FB2[POSA_FLT] FB2 --- Q1[%Q0.1.0] </pre>	<pre> CAL POSA_FLT POSA_FLT REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS LD POSA_FLT.DONE ST %Q 0.1.0 LD POSA_FLT.STAT ST FLT_STAT </pre>

POSA JOG

G3F-POAA JOG Operation	Product	GM1	GM2	GM3	GM4	GM5
	Applicable	●	●	●		

Function block	Description
<pre> graph LR subgraph "POSA_JOG" direction TB A[REQ] --- B[DONE] C[BASE] --- D[STAT] E[SLOT] F[HL] end </pre> <p>The function block diagram shows a rectangle labeled "POSA_JOG" with six pins:</p> <ul style="list-style-type: none"> REQ (input) DONE (output) BASE (input) STAT (output) SLOT (input) HL (input) 	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation ROT : Set the rotational direction of JOG operation ('0': Normal, '1': Reversal) HL : Set the high-speed/low-speed of JOG operation. ('0': Low-speed, '1': High-speed) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till REQ condition is off. STAT : Display the error code generated during the function block execution.

Function

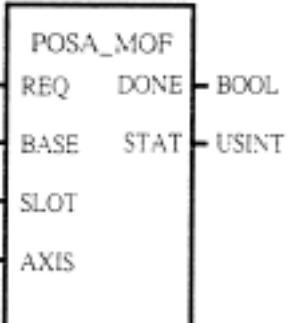
Instruction for G3F-POAA module to execute JOG operation which is used for manual operation and test operation. JOG operation is executed when REQ input is on and stopped when REQ input is off.

■ Program example

POSA MOF

G3F-POAA M Code Off

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description
 <pre> graph LR subgraph FB [POSA_MOF] direction TB R[REQ] --- I1[BOOL] D[DONE] --- O1[BOOL] B[BASE] --- I2[USINT] S[STAT] --- O2[USINT] end I1 --- P1[SLOT] I2 --- P2[USINT] I3[USINT] --- P3[AXIS] </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

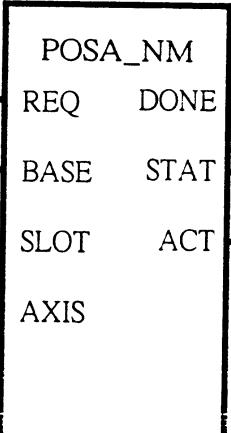
■ Function

Instruction to switch off M Code On signal which is set during positioning operation as the M code mode (with, after) in parameter.

■ Program example

POSA NM

G3F-POAA Next Move	Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●			

Function block	Description
 <pre> graph LR subgraph Inputs [Inputs] direction LR I_REQ[REQ] --- O_DONE[DONE] I_BASE[BASE] --- O_STAT[STAT] I_SLOT[SLOT] --- O_ACT[ACT] I_AXIS[AXIS] end subgraph Outputs [Outputs] direction LR O_DONE --- O_DONE O_STAT --- O_STAT O_ACT --- O_ACT O_BLANK[] end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off when ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if the location define completion signal is received from G3F-POAA and off if the instruction is used according to REQ condition.

■ Function

Instruction to operate continuously without stop from current operation speed to next operation speed.

■ Program example

Program example	LD	IL
	<pre> LD %I0.0.0 POSA_NM +-- REQ : %I0.0.0 +-- BASE : BASE +-- SLOT : SLOT +-- AXIS : AXIS +-- DONE : %Q0.1.0 +-- NM_STAT : NM_STAT +-- NM_ACT : NM_ACT </pre>	<pre> CAL POSA_NM POSA_NM REQ ? = %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS LD POSA_NM.DONE ST %Q 0.1.0 LD POSA_NM.STAT ST NM_STAT LD POSA_NM.ACT ST NM_ACT </pre>

POSA_OR

G3F-POAA Override

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description
<pre> graph LR subgraph FB [POSA_OR] direction TB IN_REQ[REQ] --> FB IN_BASE[BASE] --> FB IN_SLOT,SLOT --> FB IN_AXIS[AXIS] --> FB IN_OVR[OVR] --> FB OUT_DONE[DONE] --> OUT_DONE OUT_STAT[STAT] --> OUT_STAT OUT_ACT[ACT] --> OUT_ACT end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation OVR : Change the speed from 10 to 150% based on original operation speed and set 1~15 value(Preset value x 10% speed change) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error generated during the function block execution and the error of G3F-POAA module. ACT : On if the location order processing complete signal is received from G3F-POAA and off if the instruction is used according to REQ condition.

■ Function

Instruction to change the speed during the operation that can change the speed from 10 to 150% by 10%.

■ Program example

LD	IL
<pre> graph TD I0["%I0.0.0"] --- FB_REQ[POSA_OR] FB_REQ --- FB_DONE[DONE] FB_REQ --- FB_STAT[STAT] FB_REQ --- FB_ACT[ACT] FB_DONE --- Q010["%Q0.1.0"] FB_STAT --- OR[OR] FB_ACT --- OR OR --- OR_STAT[OR_STAT] OR --- OR_ACT[OR_ACT] </pre>	<pre> CAL POSA_OR REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS OVR := OVR LD POSA_OR.DONE ST %Q 0.1.0 LD POSA_OR.STAT ST OR_STAT LD POSA_OR.ACT ST OR_ACT </pre>

POSA_ORG

G3F-POAA zero point return	Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●			

Function block	Description
<pre> graph LR subgraph FB [POSA_ORG] direction TB R[REQ] --- FB B1[BASE] --- FB S1[SLOT] --- FB A1[AXIS] --- FB D[DONE] --- FB S2[STAT] --- FB A2[ACT] --- FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error generated during the function block execution and the error of G3F-POAA module. ACT : On if the positioning complete signal is received from G3F-POAA and off if the instruction is used according to REQ condition.

Function

Instruction to find the machine zero point as the direction and speed set by parameter. The machine zero point return operation is completed when receives the zero point return completion signal.

Program example

LD	IL
<pre> graph LR I1[%I0.0.0] --- R1[REQ] R1 --- FB1[POSA_ORG] FB1 --- D1[DONE] D1 --- Q1[%Q0.1.0] FB1 --- B1[BASE] FB1 --- S1[SLOT] FB1 --- A1[AXIS] B1 --- FB2[POSA_ORG] S1 --- FB2 A1 --- FB2 FB2 --- S2[STAT] FB2 --- A2[ACT] S2 --- ORG_STAT A2 --- ORG_ACT </pre>	<pre> CAL POSA_ORG POSA_ORG REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS LD POSA_ORG.DONE ST %Q0.1.0 LD POSA_ORG.STAT ST ORG_STAT LD POSA_ORG.ACT ST ORG_ACT </pre>

POSA_RES

G3F-POAA Error Reset

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description
<pre> graph LR subgraph FB [POSA_RES] direction TB R[REQ] --- FB B1[BASE] --- FB S1[SLOT] --- FB A1[AXIS] --- FB D1[DONE] --- FB S2[STAT] --- FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution.

Function

Instruction to reset the error generated at G3F-POPA module.

Program example

LD	IL
<pre> graph LR I0["%I0.0.0"] --- FB[POSA_RES] FB --- DONE1(()) FB --- STAT1[STAT] DONE1 --- C1(()) C1 --- RES_STAT["RES_STAT"] </pre>	<pre> CAL POSA_RES POSA_RES . REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS LD POSA_RES.DONE ST %Q 0.1.0 LD POSA_RES.STAT ST RES_STAT </pre>

POSA_RTP

G3F-POAA Return to Position	Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●			

Function block	Description
	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error generated during the function block execution and the error of G3F-POAA module. ACT : On if receives the order processing complete signal from G3F-POAA and off if the instruction is used according to REQ condition.

■ Function

Instruction to return the position before manual operation start when the position is changed by manual operation.

■ Program example

LD	IL
	<pre> CAL POSA_RTP POSA_RTP REQ := %I0_0.0 BASE := BASE SLOT := SLOT AXIS := AXIS LD POSA_RTP.DONE ST %Q 0.1.0 LD POSA_RTP.STAT ST RTP_STAT LD POSA_RTP.ACT ST RTP_ACT </pre>

POSA_SMC

G3F-POAA Next execution data number change

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description	
	Input REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation ST_SET : Change the operation data no. at next instruction within the range of 0~299. Output DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.	

■ Function

Instruction to change the operation data No. that will be processed by the next instruction within the range of 0~149.

■ Program example

LD	IL
	<pre> CAL POSA_SMC POSA_SMC REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS ST_SET := ST_SET POSA_SMC.DONE %Q0.1.0 POSA_SMC.STAT SMC_STAT </pre>

POSA_SRD

G3F-POAA Bit information reading of current operation status

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block		Description		
		Input REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base Output DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.		

[] : Indicate ARRAY variable and number in the parenthesis is the element number.

ST	ST1 content	ST2 content	ST3 content
[0]	Stop state	Synchronization operation waiting	Upper limit
[1]	Dwell state	Linear interpolation	Lower limit
[2]	Decel state	Circular interpolation	Emergency stop
[3]	Static state	Not used	Output prohibit status
[4]	Accel state	General synchronization driving	Teaching completion
[5]	Zeroing state	Ratio operation	TPB operation
[6]	Positioning state	General synchronization driving(Synchronization)	JOG low-speed operation
[7]	Interpolation state	Ratio operation	JOG high-speed operation
ST4 content	ST5 content	ST6 content	
[0]	Communicating	Not used	Dog signal(H/W)
[1]	FWD/BWD	Zero point return compensation	Zero point signal(H/W)
[2]	ZONE#1	Backlash compensation	Not used
[3]	ZONE#2	Next Move process	Upper limit(H/W)
[4]	ZONE#3	Override process	Lower limit(H/W)
[5]	Repeat operation completion	Decel stop and completion	Emergency stop(H/W)
[6]	Positioning start completion	Speed teaching completion	Not used
[7]	Not used	Speed change completion	Not used
ST7 content			
[0]	H/W Error	[3]	Busy
[1]	Error	[4]	Positioning completion
[2]	Position passing signal	[5]	M Code On
[6]		[6]	Zero point not defined
[7]		[7]	Inposition signal

Function

Function block to monitor current operation status of G3F-POAA module by bit information.

Program example

LD	IL
	CAL POSA_SRD REQ := %10.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS LD POSA_SRD.DONE ST %Q0.1.0 LD POSA_SRD.STAT ST SRD_STAT LD POSA_SRD.ST1 ST SRD_ST1 LD POSA_SRD.ST2 ST SRD_ST2 LD POSA_SRD.ST3 ST SRD_ST3 LD POSA_SRD.ST4 ST SRD_ST4 LD POSA_SRD.ST5 ST SRD_ST5 LD POSA_SRD.ST6 ST SRD_ST6 LD POSA_SRD.ST7 ST SRD_ST7

POSA_TEA

G3F-POAA Position Teaching

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description	
	Input REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation ST_SET : Set the operation data no. for position teaching.(0~149) PRESET : Set the position address for position teaching. Output DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.	

■ Function

Set the position address of certain operation data number and the zero point must have defined before.

■ Program example

LD	IL
	<pre> CAL POSA_TEA POSA_TEA . REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS ST_SET := ST_SET PRESET := PRESET LD POSA_TEA.DONE ST %Q 0.1.0 LD POSA_TEA.STAT ST TPB_STAT </pre>

POSA_TMP

Deceleration stop	Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●			

Function block	Description
<pre> graph LR subgraph FB [POSA_TMP] direction TB R[REQ] --- FB B1[BASE] --- FB S1[SLOT] --- FB A1[AXIS] --- FB D[DONE] --- FB S2[STAT] --- FB A2[ACT] --- FB S3[TMP_STAT] --- S2 A3[TMP_ACT] --- A2 end R --- FB B1 --- FB S1 --- FB A1 --- FB D --- FB S2 --- FB A2 --- FB S3 --- S2 A3 --- A2 </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off when ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G3F-POAA and off if the instruction is used according to REQ condition.

Function

Instruction to stop G3F-POAA during operation. if the function block relating stop is processed by POSP_AST, DONE condition of POSP_AST function block is not on. The machine will be operated again by toggling the function block input REQ of POSP_AST,

Program example

LD	IL
<pre> graph LR I0["%I0.0.0"] --- R1[REQ] R1 --- C1[] C1 --- Q0["%Q0.1.0"] C1 --- D1[DONE] D1 --- FB1[POSA_TMP] FB1 --- S1[STAT] FB1 --- A1[ACT] S1 --- Q0 A1 --- Q0 </pre>	<pre> CAL POSA_TMP POSA_TMP REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS LD POSA_TMP.DONE ST %Q 0.1.0 LD POSA_TMP.STAT ST TMP_STAT LD POSA_TMP.ACT ST TMP_ACT </pre>

POSA_TPB

G3F-POAA Teaching Play Back

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description		
<pre> graph LR subgraph FB [POSA_TPB] direction TB R[REQ] --- FB B1[BASE] --- FB S1[SLOT] --- FB A1[AXIS] --- FB SF[SET / FREE] --- FB D[DONE] ST[STAT] FB --> D FB --> ST end </pre>	Input REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation SET/FREE : TPB operation is available at SET status through Servo Off and is unable at Free status through Servo On so that normal operation is available.	Output DONE : On if the function block is executed without error, holds it on until function block is executed. STAT : Display the error code generated during the function block execution.	

■ Function

Place TPB SET instruction(1) to operate Teaching Play Back operation during normal operation at Servo On at initial step. Then, Servo Off status appears. After operation, place TPB Free instruction(0) for Servo On status so that normal operation is available.

■ Program example

LD	IL
<pre> graph LR I0["%I0.0.0"] --- R1[REQ] R1 --- FB1[POSA_TPB] FB1 --- D1[DONE] D1 --- Q0["%Q0.1.0"] FB1 --- B1[BASE] FB1 --- S1[SLOT] FB1 --- A1[AXIS] FB1 --- SF1[SET / FREE] B1 --- FB2[POSA_TPB] S1 --- FB2 A1 --- FB2 SF1 --- FB2 FB2 --- ST1[STAT] ST1 --- TPB_STAT </pre>	<pre> CAL POSA_TPB REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS SET / FREE := SET / FREE LD POSA_TPB.DONE ST %Q0.1.0 LD POSA_TPB.STAT ST TPB_STAT </pre>

POSA_VCG

Speed change	Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●			

Function block	Description		
	Input <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation VEL_SET: Set current speed to the speed value to be changed. Output <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off when ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G3F-POAA and off if the instruction is used according to REQ condition. 		

Function

Instruction to change the speed during operation under the static speed only. Available at single operation, repeat operation, JOG and zero point return in high-speed operation.

Program example

LD	IL
	<pre> CAL POSA_VCG POSA_VCG REQ := %I0.0.0 BASE := BASE SLOT := SLOT VEL_SET := VEL_SET LD POSA_VCG.DONE ST %Q 0.1.0 LD POSA_VCG.STAT ST VCG_STAT LD POSA_VCG.ACT ST VCG_ACT </pre>

POSA_VLT

Speed Teaching

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description	
	Input REQ : Function block execution request at rising edge BASE : G3F-POAA module installation base location number SLOT : Slot location number of G3F-POAA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation VEL_NO : Set the speed data no. for speed teaching.(0~149) VEL_SET : Set the speed value for speed teaching. (1~25,000: Pulse unit)	Output DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution. ACT : On if receives the order processing complete signal from G3F-POAA and off if the instruction is used according to REQ condition.

Function

The speed teaching instruction sets certain speed to the preset value of speed data no.

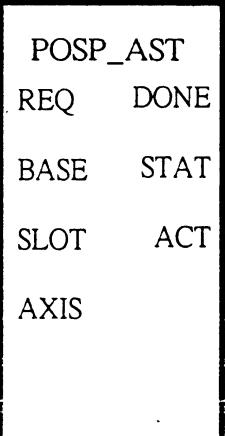
Program example

LD	IL
	<pre> CAL POSA_VLT POSA_VCG : REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS VEL_NO := VEL_NO VEL_SET := VEL_SET LD POSA_VCG.DONE ST LD POSA_VCG.STAT ST LD POSA_VCG.ACT ST </pre>

POSP AST

G3F-POPA, G4F-POPA(AXIS=0) General automatic positioning operation

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●	●	

Function block	Description
 <pre> graph LR subgraph FB [POSP_AST] direction TB R[REQ] --- FB B1[BASE] --- FB S1[SLOT] --- FB A1[AXIS] --- FB D1[DONE] --- FB S2[STAT] --- FB A2[ACT] --- FB A3[ACT] --- FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POPA or G4F-POPA module installation base location number SLOT : Slot location number of G3F-POPA or G4F-POPA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation(G3F-POPA only) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution. ACT : On if receives the positioning complete signal from G3F-POPA or G4F-POPA and off if the instruction is used according to REQ condition.

■ Function

Instruction to start the positioning operation of G3F-POPA or G4F-POPA. Check the status of G3F-POPA and G4F-POPA and use the instruction. If the positioning is completed, ACT is ON.

■ Program example

LD	IL	
<pre> LD CALL POSP_AST (%I0.0.0, BASE, SLOT, AXIS) . . . %Q0.1.0 (DONE) ENDCALL </pre>	CAL POSP_AST REQ : = %I0.0.0 BASE : = BASE SLOT : = SLOT AXIS : = AXIS LD POSP_AST.DONE ST %Q 0.1.0 LD POSP_AST.STAT ST AST_STAT LD POSP_AST.ACT ST AST_ACT	POSP_AST POSP_AST REQ : = %I0.0.0 BASE : = BASE SLOT : = SLOT AXIS : = AXIS LD POSP_AST.DONE ST %Q 0.1.0 LD POSP_AST.STAT ST AST_STAT LD POSP_AST.ACT ST AST_ACT

POSP_CRD

G3F-POPA, G4F-POPA(AXIS=0) Current status reading

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●	●	

Function block	Description
<pre> graph LR subgraph FB [POSP_CRD] direction TB FB_REQ[REQ] --- FB_DONE[DONE] FB_BASE[BASE] --- FB_STAT[STAT] FB_SLOT[SLOT] --- FB_CA[CA] FB_AXIS[AXIS] --- FB.CV[CV] FB_AXIS --- FB_MCD[MCD] FB_AXIS --- FB_CDN[CDN] end FB_REQ --- I_REQ[REQ] FB_BASE --- I_BASE[BASE] FB_SLOT --- I_SLOT[SLOT] FB_AXIS --- I_AXIS[AXIS] FB_DONE --- O_DONE[DONE] FB_STAT --- O_STAT[STAT] FB_CA --- O_CA[CA] FB.CV --- O.CV[CV] FB_MCD --- O.MCD[MCD] FB_CDN --- O.CDN[CDN] </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POPA or G4F-POPA module installation base location number SLOT : Slot location number of G3F-POPA or G4F-POPA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation(G3F-POPA only) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during the function block execution. CA : Current position address display CV : Current operation speed display MCD : Current M Code value display CDN : Current operation data No. display

Function

Function block to monitor current operation status of G3F-POPA or G4F-POPA modules.

Program example

LD	IL
<pre> LD %I0.0.0 --- POSP_CRD[POSP_CRD] POSP_CRD --- %Q0.1.0 POSP_CRD --- CRD_STAT[CRD_STAT] POSP_CRD --- CRD_CA[CRD_CA] POSP_CRD --- CRD.CV[CRD.CV] POSP_CRD --- CRD_MCD[CRD_MCD] POSP_CRD --- CRD_CDN[CRD_CDN] </pre>	<pre> IL CAL POSP_CRD POSP_CRD REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS POSP_CRD.DONE := %Q0.1.0 POSP_CRD.STAT := CRD_STAT POSP_CRD.CA := CRD_CA POSP_CRD.CV := CRD.CV POSP_CRD.MCD := CRD_MCD POSP_CRD.CDN := CRD_CDN </pre>

POSP_EMG

G3F-POPA, G4F-POPA(AXIS=0) Emergency stop

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●	●	

Function block	Description
<pre> graph LR subgraph FB [POSP_EMG] direction TB REQ[REQ] --- FB BASE[BASE] --- FB SLOT[SLOT] --- FB DONE[DONE] --- FB STAT[STAT] --- FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POPA or G4F-POPA module installation base location number SLOT : Slot location number of G3F-POPA or G4F-POPA module installation base <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error. STAT : Display the error code generated during the function block execution.

Function

Execute emergency stop during operating G3F-POPA or G4F-POPA module. Emergency stop is separated with external emergency stop by external signal and internal emergency stop by function block. In case of internal emergency stop, external emergency stop signal shall be always B contact(NC: Normal close) status. This instruction has priority for other function block.

Program example

LD	IL
<pre> LD %I0.0.0 --- ---- POSP_EMG ---- REQ ---- BASE ---- SLOT ---- DONE --- ---- %Q0.1.0 ----()--- EMG_STAT ---- STAT </pre>	<pre> CAL POSP_EMG POSP_EMG REQ := %I0.0.0 BASE := BASE SLOT := SLOT LD POSP_EMG.DONE ST %Q 0.1.0 LD POSP_EMG.STAT ST EMG_STAT </pre>

POSP_FLT

G3F-POPA, G4F-POPA(AXIS=0) Floating zero point set

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●	●	

Function block	Description
<pre> graph LR subgraph FB [POSP_FLT] direction TB REQ[REQ] --> FB BASE[BASE] --> FB SLOT[SLOT] --> FB AXIS[AXIS] --> FB DONE[DONE] --> FB STAT[STAT] --> FB end </pre>	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POPA and G4F-POPA module installation base location number SLOT : Slot location number of G3F-POPA and G4F-POPA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation(G3F-POPA only) <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and hold on till next function block is executed. STAT : Display the error code generated during the function block execution.

Function

Instruction to use current position as zero point compulsively instead of machine zero point. Current position will be the value of zero point address at the parameter.

Program example

LD	IL
<pre> LD %I0.0.0 --- ---- POSP_FLT ---- REQ BASE --- ---- POSP_FLT ---- DONE --- ---- (%Q0.1.0) SLOT --- ---- POSP_FLT ---- STAT --- ---- FLT_STAT AXIS --- ---- POSP_FLT </pre>	<pre> IL CAL POSP_FLT POSP_FLT REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS LD POSP_FLT.DONE ST %Q0.1.0 LD POSP_FLT.STAT ST EMG_STAT </pre>

POSP_INC

G3F-POPA, G4F-POPA(AXIS=0) Inchng operation

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●	●	

Function block	Description	
<pre> graph LR subgraph POSP_INC [POSA_INC] direction TB REQ --- POSP_INC[POSA_INC] BASE --- POSP_INC SLOT --- POSP_INC AXIS --- POSP_INC ROT --- POSP_INC INCH_AMT --- POSP_INC POSP_INC --- DONE POSP_INC --- STAT POSP_INC --- ACT end </pre>	Input <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POPA or G4F-POPA module installation base location number SLOT : Slot location number of G3F-POPA or G4F-POPA module installation base AXIS : 0: X-axis operation, 1: Y-axis operation(G3F-POPA only) ROT : Set the rotational direction of inching operation ('0': FWD, '1': BWD) INCH_AMT : Set the pulse amount to output (1~99) 	Output <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT : Display the error code generated during execution. ACT : On if the order processing complete signal is received from G3F-POPA or G4F-POPA and off if the instruction is used according to REQ condition.

Function

As the manual operation, output the pulse of the value set by rotational direction and INCH_AMT whenever REQ Input signal is toggled.

Program example

LD	IL
<pre> LD %I0.0 --- POSP_INC[POSP_INC] POSP_INC --- %Q0.1.0 POSP_INC --- REQ POSP_INC --- BASE POSP_INC --- SLOT POSP_INC --- AXIS POSP_INC --- ROT POSP_INC --- INCH_AMT POSP_INC --- DONE POSP_INC --- STAT POSP_INC --- ACT </pre>	<pre> CAL POSP_INC POSP_INC REQ := %I0.0.0 BASE := BASE SLOT := SLOT AXIS := AXIS ROT := ROT INCH_AMT := INCH_AMT LD POSP_INC.DONE POSP_INC.DONE ST %Q 0.1.0 %Q 0.1.0 LD POSP_INC.STAT POSP_INC.STAT ST INC_STAT INC_STAT LD POSP_INC.ACT POSP_INC.ACT ST INC_ACT INC_ACT </pre>

POSP_INT

G3F-POPA Linear interpolation

Product	GM1	GM2	GM3	GM4	GM5
Applicable	●	●	●		

Function block	Description
	<p>Input</p> <ul style="list-style-type: none"> REQ : Function block execution request at rising edge BASE : G3F-POPA module installation base location number SLOT : Slot location number of G3F-POPA module installation base <p>Output</p> <ul style="list-style-type: none"> DONE : On if the function block is executed without error and off if ACT is on. STAT_X: Display the error of the function block or the error of G3F-POPA X axis. STAT_Y: Display the error of the function block or the error of G3F-POPA Y axis. ACT : On if the positioning completion signal is received from G3F-POPA and off if the instruction is used according to REQ condition.

■ Function

Instruction for 2-axis's linear interpolation.

■ Program example

LD	IL
	<pre> CAL POSP_INT POSP_INT REQ := %I0.0 BASE := BASE SLOT := SLOT LD POSP_INT.DONE %Q 0.1.0 ST POSP_INT.STAT LD INT_STAT_X ST POSP_INT.STAT LD INT_STAT_Y ST POSP_INT.ACT LD INT_ACT </pre>