

Motion Controller

**GM1 Series**

# **Reference Manual**

---

Instruction Edition

(MEMO)

## Introduction

Thank you for purchasing a Panasonic product. Before you use the product, please carefully read through the installation instructions and the manuals, and understand them in detail to use the product properly.

## Types of Manual

- There are different types of manuals for the GM1 series, as listed below. Please refer to a relevant manual for the unit and purpose of your use.
- The manuals can be downloaded from our website: <https://industrial.panasonic.com/ac/e/motor/motion-controller/mc/gm1/index.jsp>.

### Manuals for GM1 series

| Manual name   | Manual code   | Description  |
|---|---------------|--|
| GM1 Controller RTEX User's Manual (Setup Edition)         | WUME-GM1RTXSU | Explains wiring between the GM1 and its peripheral devices, installation method, and operation check method.                     |
| GM1 Controller EtherCAT User's Manual (Setup Edition)     | WUME-GM1ETCSU |  |
| GM1 Controller RTEX User's Manual (Operation Edition)     | WUME-GM1RTXOP | Explains how to use GM Programmer and PANATERM Lite for GM, set up each function, create projects, and perform other operations. |
| GM1 Controller EtherCAT User's Manual (Operation Edition) | WUME-GM1ETCOP |  |
| GM1 Series Reference Manual (Hardware Edition)            | WUME-GM1H     | Explains the functions and performance of each GM1 unit.   |
| GM1 Series Reference Manual (Instruction Edition)         | WUME-GM1PGR   | Explains the specifications of each instruction that can be used with the GM1 Series.  |
| GM1 Series Reference Manual (Analog I/O Unit)             | WUME-GM1AIO   | Explains the functions and performance of the GM1 Analog Expansion Unit.   |
| GM1 Series Reference Manual (Pulse Output Unit)           | WUME-GM1PG    | Explains the functions and performance of the GM1 Pulse Output Unit.   |

## Copyright / Trademarks

- The copyright of this manual belongs to **Panasonic Industry Co., Ltd.**
- Unauthorized reproduction of this manual is strictly prohibited.
- Windows is a registered trademark of Microsoft Corporation in the U.S. and other countries.
- Ethernet is a registered trademark of Fuji Xerox Co., Ltd. and Xerox Corporation.
- EtherCAT is a registered trademark of and patented technology licensed by Beckhoff Automation GmbH, Germany.
- EtherNet/IP is a registered trademark of ODVA (Open DeviceNet Vendor Association).
- SDHC and SD logos are trademarks of LLC.
- Other company and product names are trademarks or registered trademarks of their respective companies.

(MEMO)

# Table of Contents

|   |            |
|---|------------|
| <b>1 List of Instructions</b> .....   | <b>1-1</b> |
| 1.1 List of Ladder Instructions.....  | 1-2        |
| 1.2 List of Function Instructions.....  | 1-3        |
| 1.3 List of Function Block Instructions.....                                  | 1-9        |
| 1.3.1 Basic Instructions.....   | 1-9        |
| 1.3.2 Motion Control Function Blocks (Single Axis Control).....               | 1-11       |
| 1.3.3 Motion Control Function Blocks (Synchronous Control).....               | 1-14       |
| 1.3.4 Motion Control Function Blocks (Multi-axis Control).....                | 1-15       |
| 1.3.5 Motion Control Function Blocks (Motion Communication Control) ..        | 1-15       |
| 1.3.6 Motion Control Function Blocks (Auxiliary Function).....                | 1-18       |
| 1.3.7 Function Blocks (Others).....   | 1-20       |
| 1.3.8 Function Blocks (For the GM1 Pulse Output Unit).....                    | 1-27       |
| 1.4 List of Function Block Instructions that Cannot Be Used with the GM1..... | 1-29       |
| <b>2 Ladder Instructions</b> .....  | <b>2-1</b> |
| 2.1 Ladder Instructions.....  | 2-2        |
| 2.1.1 NO Contact.....   | 2-2        |
| 2.1.2 NC Contact.....   | 2-3        |
| 2.1.3 Rising Edge Detection Contact.....                                      | 2-3        |
| 2.1.4 Falling Edge Detection Contact.....                                     | 2-4        |
| 2.1.5 Parallel NO Contact.....  | 2-5        |
| 2.1.6 Parallel NC Contact.....  | 2-6        |
| 2.1.7 Coil.....   | 2-7        |
| 2.1.8 Negated Coil.....   | 2-7        |
| 2.1.9 Set Coil.....   | 2-8        |
| 2.1.10 Reset Coil.....  | 2-9        |
| <b>3 Functions</b> .....  | <b>3-1</b> |
| 3.1 Substitution Instruction.....   | 3-3        |
| 3.1.1 MOVE (Substitution).....  | 3-3        |
| 3.2 Arithmetic Operation Instructions.....                                    | 3-4        |
| 3.2.1 ADD (Addition).....   | 3-4        |
| 3.2.2 SUB (Subtraction).....  | 3-5        |
| 3.2.3 MUL (Multiplication).....   | 3-6        |
| 3.2.4 DIV (Division).....   | 3-7        |
| 3.2.5 MOD (Remainder).....  | 3-8        |
| 3.3 Boolean Operation Instructions.....                                       | 3-9        |
| 3.3.1 AND (Logical AND).....  | 3-9        |
| 3.3.2 OR (Logical OR).....  | 3-10       |
| 3.3.3 NOT (Negation).....   | 3-10       |
| 3.3.4 XOR (Exclusive OR).....   | 3-11       |
| 3.3.5 AND_THEN (Logical AND).....   | 3-12       |
| 3.3.6 OR_ELSE (Logical OR).....   | 3-13       |
| 3.4 Comparison Operation Instructions.....                                    | 3-15       |
| 3.4.1 EQ ("Equal" Comparison).....  | 3-15       |

|        |  |      |
|--------|--|------|
| 3.4.2  | NE (“Not Equal” Comparison)                                | 3-15 |
| 3.4.3  | LT (“Less Than” Comparison)                                | 3-16 |
| 3.4.4  | LE (“Less Than or Equal” Comparison)                       | 3-17 |
| 3.4.5  | GT (“Greater Than” Comparison)                             | 3-18 |
| 3.4.6  | GE (“Greater Than Or Equal” Comparison)                    | 3-19 |
| 3.5    | Bit Shift Instructions                                     | 3-21 |
| 3.5.1  | SHL (Shift Left)   | 3-21 |
| 3.5.2  | SHR (Shift Right)  | 3-22 |
| 3.5.3  | ROL (Rotate Left)  | 3-22 |
| 3.5.4  | ROR (Rotate Right)   | 3-23 |
| 3.6    | Selection Instructions                                     | 3-25 |
| 3.6.1  | SEL (Binary Selector)                                      | 3-25 |
| 3.6.2  | MUX (Multiplexer)  | 3-26 |
| 3.6.3  | LIMIT (Limiter)  | 3-27 |
| 3.6.4  | MAX (Maximum Value)  | 3-28 |
| 3.6.5  | MIN (Minimum Value)  | 3-28 |
| 3.7    | Numerical Operation Instructions                           | 3-30 |
| 3.7.1  | ABS (Absolute Value)                                       | 3-30 |
| 3.7.2  | SQRT (Square Root)   | 3-30 |
| 3.7.3  | LN (Natural Logarithm)                                     | 3-31 |
| 3.7.4  | LOG (Common Logarithm)                                     | 3-32 |
| 3.7.5  | EXP (Natural Exponent)                                     | 3-33 |
| 3.7.6  | EXPT (Exponentiation)                                      | 3-33 |
| 3.7.7  | SIN (Trigonometric Function Sine)                          | 3-34 |
| 3.7.8  | COS (Trigonometric Function Cosine)                        | 3-35 |
| 3.7.9  | TAN (Trigonometric Function Tangent)                       | 3-36 |
| 3.7.10 | ASIN (Trigonometric Function Arc Sine)                     | 3-37 |
| 3.7.11 | ACOS (Trigonometric Function Arc Cosine)                   | 3-38 |
| 3.7.12 | ATAN (Trigonometric Function Arc Tangent)                  | 3-38 |
| 3.8    | Data Type Conversion Instructions                          | 3-40 |
| 3.8.1  | "Type 1" _TO_ "Type 2" ("Type 1">"Type 2" Conversion)      | 3-40 |
| 3.8.2  | TRUNC (Real Number to DINT Conversion)                     | 3-41 |
| 3.8.3  | TRUNC_INT (Real Number to INT Conversion)                  | 3-41 |
| 3.9    | Character string instructions                              | 3-43 |
| 3.9.1  | LEN (Length of a Character String)                         | 3-43 |
| 3.9.2  | LEFT (Extracting Characters from the Left End)             | 3-43 |
| 3.9.3  | RIGHT (Extracting Characters from the Right End)           | 3-44 |
| 3.9.4  | MID (Extracting Characters from the Specified Position)    | 3-45 |
| 3.9.5  | CONCAT (Concatenating Character Strings)                   | 3-46 |
| 3.9.6  | INSERT (Inserting a Character String)                      | 3-47 |
| 3.9.7  | DELETE (Deleting a Character String)                       | 3-48 |
| 3.9.8  | REPLACE (Replacing a Character String)                     | 3-49 |
| 3.9.9  | FIND (Search for Character String)                         | 3-50 |
| 3.10   | Other Instructions   | 3-52 |
| 3.10.1 | SIZEOF (Get the Size)                                      | 3-52 |
| 3.10.2 | ADR (Get the Address)                                      | 3-52 |
| 3.11   | SD Memory Card Slot Instruction                            | 3-54 |
| 3.11.1 | SYS_GetSDCoverState (Get SD Card Cover Open / Close State) | 3-54 |
| 3.11.2 | SYS_GetSDAccessRdy (Get SD Card Access Ready State)        | 3-54 |

|  |            |
|--|------------|
| <b>4 Function Blocks (Basic Instructions)</b> .....                              | <b>4-1</b> |
| 4.1 Timer Instructions .....   | 4-2        |
| 4.1.1 TON (Timer ON).....  | 4-2        |
| 4.1.2 TOF (Timer OFF) .....  | 4-3        |
| 4.1.3 TP (Timer Pulse).....  | 4-4        |
| 4.1.4 RTC (Realtime Clock).....  | 4-6        |
| 4.2 Counter Instructions.....  | 4-7        |
| 4.2.1 CTU (Up Counter).....  | 4-7        |
| 4.2.2 CTD (Down Counter) .....   | 4-8        |
| 4.2.3 CTUD (Up-down Counter) .....   | 4-9        |
| 4.3 Edge Detection Instructions.....   | 4-11       |
| 4.3.1 R_TRIG (Rising Edge Detection).....  | 4-11       |
| 4.3.2 F_TRIG (Falling Edge Detection).....                                       | 4-11       |
| 4.4 Bistable Circuit Instructions.....   | 4-13       |
| 4.4.1 SR (Set-priority Bistable Circuit) .....                                   | 4-13       |
| 4.4.2 RS (Reset-priority Bistable Circuit) .....                                 | 4-14       |
| <br>   |            |
| <b>5 Motion Control Function Blocks (Single Axis Control)</b> .....              | <b>5-1</b> |
| 5.1 Servo ON .....   | 5-2        |
| 5.1.1 MC_Power (Servo ON) .....  | 5-2        |
| 5.2 Home Return.....   | 5-4        |
| 5.2.1 PMC_Home (Home Return).....  | 5-4        |
| 5.2.2 MC_Home (Home Return) .....  | 5-6        |
| 5.3 Control Switch.....  | 5-8        |
| 5.3.1 SMC_SetControllerMode (Control Mode Setting).....                          | 5-8        |
| 5.4 Stop.....  | 5-9        |
| 5.4.1 MC_Stop (Forced Stop) .....  | 5-9        |
| 5.4.2 MC_Halt (Halt) .....   | 5-10       |
| 5.5 JOG / Inching .....  | 5-11       |
| 5.5.1 MC_Jog (Jogging).....  | 5-11       |
| 5.5.2 SMC_Inch (Inching) .....   | 5-12       |
| 5.6 Position Control.....  | 5-14       |
| 5.6.1 MC_MoveAbsolute (Absolute Value Positioning).....                          | 5-14       |
| 5.6.2 MC_MoveRelative (Relative Value Positioning).....                          | 5-15       |
| 5.6.3 MC_MoveAdditive (Target Position Change) .....                             | 5-16       |
| 5.6.4 MC_MoveSuperImposed (Superimposed positioning) .....                       | 5-18       |
| 5.6.5 MC_PositionProfile (Position Profile Move) .....                           | 5-20       |
| 5.6.6 Default Setting for Variables of the MC_TP_REF Type Structure....          | 5-22       |
| 5.6.7 SMC_MoveContinuousAbsolute (Absolute Value Position Velocity<br>Move)..... | 5-24       |
| 5.6.8 SMC_MoveContinuousRelative (Relative Value Position Velocity<br>Move)..... | 5-25       |
| 5.7 Velocity Control .....   | 5-27       |
| 5.7.1 MC_MoveVelocity (Velocity Control).....                                    | 5-27       |
| 5.7.2 MC_VelocityProfile (Velocity Profile Movement) .....                       | 5-28       |
| 5.7.3 MC_AccelerationProfile (Acceleration Profile Movement) .....               | 5-29       |
| 5.8 Torque Control .....   | 5-32       |
| 5.8.1 PMC_SetTorque (Torque Control).....  | 5-32       |
| 5.8.2 SMC_SetTorque (Torque Control).....  | 5-33       |

|   |            |
|---|------------|
| <b>6 Motion Control Function Blocks (Synchronous Control)</b>                       | <b>6-1</b> |
| 6.1 Cam Operation   | 6-2        |
| 6.1.1 MC_CamIn (Start Cam Operation)  | 6-2        |
| 6.1.2 MC_CamOut (Cancel Cam Operation)  | 6-5        |
| 6.1.3 MC_CamTableSelect (Select Cam Table)  | 6-6        |
| 6.1.4 SMC_GetTappetValue (Tappet Output)  | 6-8        |
| 6.2 Gear Operation  | 6-11       |
| 6.2.1 MC_GearIn (Start Gear Operation)  | 6-11       |
| 6.2.2 MC_GearInPos (Position Specified Gear Operation)                              | 6-12       |
| 6.2.3 MC_GearOut (Cancel Gear Operation)  | 6-14       |
| 6.3 Phase Correction  | 6-16       |
| 6.3.1 MC_Phasing (Master Axis Phase Correction)                                     | 6-16       |
| <b>7 Motion Control Function Blocks (Multi-axis Control)</b>                        | <b>7-1</b> |
| 7.1 Interpolation Control   | 7-2        |
| 7.1.1 PMC_Interpolator2D (2-axis Interpolation Control)                             | 7-2        |
| 7.1.2 PMC_Interpolator3D (3-axis Interpolation Control)                             | 7-3        |
| 7.1.3 PMC_NCDecoder (CNC Table Conversion)  | 7-5        |
| <b>8 Motion Control Function Blocks (Motion Communication Control)</b>              | <b>8-1</b> |
| 8.1 RTEX  | 8-2        |
| 8.1.1 Types of Data To Be Handled by AMP Function Blocks                            | 8-2        |
| 8.1.2 RTEX_ClearAmpAlarm (Clear Amplifier Alarm)                                    | 8-2        |
| 8.1.3 RTEX_ReadAmpAlarm (Read Amplifier Alarm)                                      | 8-4        |
| 8.1.4 RTEX_ReadAmpState (Amplifier Alarm Status)                                    | 8-5        |
| 8.1.5 RTEX_ReadAmpData (Amplifier Monitor)  | 8-6        |
| 8.1.6 RTEX_ReadAmpParameter (Read Amplifier Parameter)                              | 8-7        |
| 8.1.7 RTEX_WriteAmpParameter (Write Amplifier Parameter)                            | 8-8        |
| 8.1.8 RTEX_WriteAmpEEPROM (Write Amplifier EEPROM)                                  | 8-9        |
| 8.1.9 RTEX_Reset (Reset RTEX)   | 8-9        |
| 8.1.10 RTEX_ClearAmpMultiTurnData (Clear Amplifier Multi-turn Data)                 | 8-10       |
| 8.1.11 RTEX_ClearAmpPositionalDeviation (Clear Amplifier Deviation Counter)         | 8-11       |
| 8.1.12 RTEX_GetTrackingCommandError (Read RTEX Command Send Statistics Information) | 8-12       |
| 8.1.13 RTEX_ReadPot (Read POT of Amplifier)   | 8-13       |
| 8.1.14 RTEX_ReadNot (Read NOT of Amplifier)   | 8-14       |
| 8.2 EtherCAT  | 8-15       |
| 8.2.1 ETC_CO_SdoRead (Read Slave Parameter)   | 8-15       |
| 8.2.2 ETC_CO_SdoRead4 (Read Four Bytes of Slave Parameter)                          | 8-16       |
| 8.2.3 ETC_CO_SdoReadDWord (Read Double Word of Slave Parameter)                     | 8-17       |
| 8.2.4 ETC_CO_SdoRead_Access (Read Slave Parameter Index)                            | 8-18       |
| 8.2.5 ETC_CO_SdoRead_Channel (Read Priority Specification of Slave Parameter)       | 8-19       |
| 8.2.6 ETC_CO_SdoWrite (Write Slave Parameter)                                       | 8-21       |
| 8.2.7 ETC_CO_SdoWrite4 (Write Four Bytes of Slave Parameter)                        | 8-22       |
| 8.2.8 ETC_CO_SdoWriteDWord (Write Double Words of Slave Parameter)                  | 8-24       |
| 8.2.9 ETC_CO_SdoWrite_Access (Write Slave Parameter Index)                          | 8-25       |
| 8.2.10 ReadIdentification (Read Slave Identification Data)                          | 8-26       |



|           |  |             |
|-----------|--|-------------|
| 8.2.11    | ReadMemory (Read Slave Memory).....  | 8-27        |
| 8.2.12    | ReadNbrSlaves (Read the Number of Connected Slaves).....                     | 8-28        |
| 8.2.13    | WriteMemory (Write Slave Memory).....  | 8-29        |
| 8.2.14    | PETC_ClearAmpPositionalDeviation (Clear Amplifier Deviation Counter).....    | 8-30        |
| 8.2.15    | IoDrvEtherCAT.GetStatistics (Get EtherCAT Frame Statistics Information)..... | 8-31        |
| <b>9</b>  | <b>Motion Control Function Blocks (Auxiliary Function).....</b>              | <b>9-1</b>  |
| 9.1       | Motion Auxiliary Function (Monitoring).....                                  | 9-2         |
| 9.1.1     | MC_ReadActualPosition (Read Current Position) .....                          | 9-2         |
| 9.1.2     | MC_ReadActualVelocity (Read Current Velocity) .....                          | 9-2         |
| 9.1.3     | PMC_ReadActualTorque (Read Current Torque) .....                             | 9-3         |
| 9.1.4     | MC_ReadActualTorque (Read Current Torque).....                               | 9-4         |
| 9.1.5     | MC_ReadAxisError (Read Axis Error) .....                                     | 9-4         |
| 9.1.6     | MC_ReadStatus (Read Status) .....  | 9-5         |
| 9.1.7     | SMC_InPosition (In-position Judgment) .....                                  | 9-7         |
| 9.1.8     | SMC_ReadFBError (Read Oldest Error) .....                                    | 9-8         |
| 9.1.9     | SMC_ClearFBError (Clear Oldest Error) .....                                  | 9-9         |
| 9.1.10    | SMC_CheckAxisCommunication (Check Axis Communication Status).....            | 9-10        |
| 9.1.11    | SMC_CheckLimits (Check Exceeding Limits).....                                | 9-11        |
| 9.1.12    | SMC_GetMaxSetAccDec (Measure Maximum Acceleration / Deceleration).....       | 9-12        |
| 9.1.13    | SMC_GetMaxSetVelocity (Measure Maximum Velocity) .....                       | 9-12        |
| 9.1.14    | SMC_GetTrackingError (Measure Tracking Error).....                           | 9-13        |
| 9.1.15    | SMC_MeasureDistance (Measure Turnaround Travel Distance) ..                  | 9-14        |
| 9.1.16    | SMC_ReadSetPosition (Read Axis Set Position) .....                           | 9-15        |
| 9.2       | Motion Auxiliary Function (Change / Reset) .....                             | 9-16        |
| 9.2.1     | MC_Reset (Axis Error Reset) .....  | 9-16        |
| 9.2.2     | SMC3_ReinitDrive (Reinitialize Axis).....                                    | 9-16        |
| 9.2.3     | MC_SetPosition (Change Current Position) .....                               | 9-17        |
| 9.3       | Motion Auxiliary Function (Other Functions).....                             | 9-18        |
| 9.3.1     | PMC_ReadLatchPosition (Amplifier Latch Monitor).....                         | 9-18        |
| 9.3.2     | PMC_StopLatchPosition (Stop Amplifier Latch) .....                           | 9-20        |
| 9.3.3     | MC_TouchProbe (Enable AMP Latch Monitoring) .....                            | 9-23        |
| 9.3.4     | MC_AbortTrigger (Disable AMP Latch Monitoring).....                          | 9-25        |
| 9.3.5     | MC_DigitalCamSwitch (Enable Digital Cam Switch) .....                        | 9-26        |
| 9.3.6     | SMC_BacklashCompensation (Compensate Backlash).....                          | 9-29        |
| <b>10</b> | <b>Other Function Blocks .....</b>   | <b>10-1</b> |
| 10.1      | COM Port (General-purpose Communication).....                                | 10-4        |
| 10.1.1    | COM.Open (Open COM port) .....   | 10-4        |
| 10.1.2    | COM.Close (Close COM Port).....  | 10-6        |
| 10.1.3    | COM.Read (Read COM Port) .....   | 10-7        |
| 10.1.4    | COM.Write (Write COM Port).....  | 10-8        |
| 10.1.5    | COM.ERROR (Error ID).....  | 10-8        |
| 10.2      | COM port (Modbus COM).....   | 10-10       |
| 10.2.1    | IoDrvModbusComPort .....   | 10-10       |
| 10.2.2    | IoDrvModbus.ModbusChannel(Start Sending Modbus Command)                      | 10-10       |
| 10.2.3    | IoDrvModbus.ModbusRequest (Modbus Request).....                              | 10-11       |

|         |   |       |
|---------|---|-------|
| 10.2.4  | IoDrvModbus.ModbusRequest 2 (Modbus Request 2)                            | 10-12 |
| 10.2.5  | IoDrvModbus.ModbusSlaveComPort  | 10-14 |
| 10.2.6  | IoDrvModbus.MB_ErrorCodes (Error Codes)                                   | 10-14 |
| 10.3    | LAN port (IoDrvEthernet)  | 10-16 |
| 10.3.1  | IoDrvEthernet   | 10-16 |
| 10.3.2  | IoDrvEthernet.IPARRAY_TO_INADDR (Array Type to Union Type)                | 10-16 |
| 10.3.3  | IoDrvEthernet.IPARRAY_TO_IPSTRING (Array Type to Character String Type)   | 10-17 |
| 10.3.4  | IoDrvEthernet.IPARRAY_TO_UDINT (Array Type to UDINT Type)                 | 10-17 |
| 10.3.5  | IoDrvEthernet.IPSTRING_TO_UDINT (Character String Type to UDINT Type)     | 10-18 |
| 10.3.6  | IoDrvEthernet.UDINT_TO_IPARRAY (UDINT Type to Array Type)                 | 10-18 |
| 10.3.7  | IoDrvEthernet.UDINT_TO_IPSTRING (UDINT Type to Character String Type)     | 10-18 |
| 10.4    | LAN Port (General-purpose Communication)                                  | 10-20 |
| 10.4.1  | NBS.TCP_Client (Connect to TCP Client)                                    | 10-20 |
| 10.4.2  | NBS.TCP_Connection (Connect TCP)  | 10-20 |
| 10.4.3  | NBS.TCP_Read (Receive TCP Data)   | 10-21 |
| 10.4.4  | NBS.TCP_Server (Connect TCP Server)                                       | 10-22 |
| 10.4.5  | NBS.TCP_Write (Send TCP Data)   | 10-23 |
| 10.4.6  | NBS.UDP_Peer (Open UDP Port)  | 10-23 |
| 10.4.7  | NBS.UDP_Receive (Receive UDP Data)  | 10-24 |
| 10.4.8  | NBS.ERROR (Error Code)  | 10-25 |
| 10.4.9  | NBS.UDP_Send (Send UDP Data)  | 10-26 |
| 10.5    | LAN Port (Modbus TCP)   | 10-28 |
| 10.5.1  | IoDrvModbusTCP  | 10-28 |
| 10.5.2  | IoDrvModbusTCP.ModbusChannel (Start Sending Modbus Command)               | 10-28 |
| 10.5.3  | IoDrvModbusTCP.ModbusRequest (Modbus Request)                             | 10-29 |
| 10.5.4  | IoDrvModbusTCPSlave   | 10-30 |
| 10.5.5  | IoDrvModbus.MB_ErrorCodes (Error Codes)                                   | 10-31 |
| 10.6    | LAN Port (EtherNet/IP)  | 10-33 |
| 10.6.1  | IoDrvEtherNetIP (EtherNet/IP Scanner Device)                              | 10-33 |
| 10.6.2  | RemoteAdapter (Remote Adapter Device)                                     | 10-34 |
| 10.6.3  | IoDrvEtherNetIPAdapter (EtherNet/IP adapter device)                       | 10-36 |
| 10.6.4  | Module (EtherNet/IP Module Device)  | 10-37 |
| 10.6.5  | Apply_Attributes (Apply_Attributes Service)                               | 10-38 |
| 10.6.6  | Generic_Service (Generic Service Execution)                               | 10-39 |
| 10.6.7  | Get_Attribute_Single (Inquire Specific Attributes of a Specific Instance) | 10-40 |
| 10.6.8  | Get_Attributes_All (Inquire All Attributes of a Specific Instance)        | 10-41 |
| 10.6.9  | Set_Attribute_Single (Set Specific Attributes of a Specific Instance)     | 10-42 |
| 10.6.10 | Set_Attributes_All (Set All Attributes of a Specific Instance)            | 10-43 |
| 10.6.11 | NOP (NOP Service)   | 10-44 |
| 10.6.12 | Reset (Reset Service)   | 10-45 |
| 10.6.13 | Start (Start Service)   | 10-45 |
| 10.6.14 | Stop (Stop Service)   | 10-46 |
| 10.6.15 | ENIP.ERROR (Message Service Instruction Error Code)                       | 10-47 |
| 10.6.16 | ENIP.CIPClass (Service Class Code)  | 10-49 |
| 10.7    | SD Card Operation (File Operation)  | 10-52 |

|           |  |             |
|-----------|--|-------------|
| 10.7.1    | FILE.Open (Open File)  | 10-52       |
| 10.7.2    | FILE.Close (Close File)  | 10-53       |
| 10.7.3    | FILE.Read (Read File)  | 10-53       |
| 10.7.4    | FILE.Write (Write File)  | 10-54       |
| 10.7.5    | FILE.Flush (Flush File)  | 10-55       |
| 10.7.6    | FILE.Copy (Copy File)  | 10-56       |
| 10.7.7    | FILE.Rename (Rename File)  | 10-57       |
| 10.7.8    | FILE.Delete (Delete File)  | 10-58       |
| 10.7.9    | FILE.EOF (End of File)   | 10-58       |
| 10.7.10   | FILE.GetAttribute (Get File Attribute)                           | 10-59       |
| 10.7.11   | FILE.GetPos (Get File Offset)                                    | 10-60       |
| 10.7.12   | FILE.GetSize (Get File Size)                                     | 10-60       |
| 10.7.13   | FILE.GetTime (Get File Update Time)                              | 10-61       |
| 10.7.14   | FILE.SetPos (Set File Offset)                                    | 10-62       |
| 10.7.15   | FILE.ERROR (Error ID)  | 10-62       |
| 10.8      | SD Card Operation (Directory Operation)                          | 10-64       |
| 10.8.1    | FILE.DirCreate (Create Directory)                                | 10-64       |
| 10.8.2    | FILE.DirOpen (Open Directory)                                    | 10-64       |
| 10.8.3    | FILE.DirClose (Close Directory)                                  | 10-65       |
| 10.8.4    | FILE.DirCopy (Copy Directory)                                    | 10-66       |
| 10.8.5    | FILE.DirRename (Rename Directory)                                | 10-67       |
| 10.8.6    | FILE.DirRemove (Delete Directory)                                | 10-67       |
| 10.8.7    | FILE.DirList (Directory List)                                    | 10-68       |
| 10.9      | Clock Setting  | 10-70       |
| 10.9.1    | SYS_GetTime (Get Time)   | 10-70       |
| 10.9.2    | SYS_SetTime (Set Time)   | 10-70       |
| 10.9.3    | SYS_GetTimezone (Get Time Zone Information)                      | 10-71       |
| 10.9.4    | SYS_SetTimezone (Set Time Zone Information)                      | 10-71       |
| 10.9.5    | SYS_DateConcat (Convert from UINT Type to DATE Type)             | 10-72       |
| 10.9.6    | SYS_DateSplit (Convert from DATE Type to UINT Type)              | 10-73       |
| 10.9.7    | SYS_DTConcat (Convert from UINT Type to DT Type)                 | 10-73       |
| 10.9.8    | SYS_DTSplit (Convert from UINT Type to DT Type)                  | 10-74       |
| 10.9.9    | SYS_GetDayOfWeek (Get Day of the Week)                           | 10-75       |
| 10.9.10   | SYS_TODConcat (Convert from UINT Type to TOD Type)               | 10-75       |
| 10.9.11   | SYS_TODSplit (Convert from TOD Type to UINT Type)                | 10-76       |
| 10.9.12   | ERROR (Clock Instruction Error Code)                             | 10-77       |
| 10.10     | System Data  | 10-78       |
| 10.10.1   | SYS_GetSystemError (Get System Error)                            | 10-78       |
| 10.11     | PID Control  | 10-79       |
| 10.11.1   | PD (PD Control)  | 10-79       |
| 10.11.2   | PID (PID Control)  | 10-80       |
| 10.11.3   | PID_FIXCYCLE [PID Control (Any Cycle Time)]                      | 10-81       |
| <b>11</b> | <b>Function Blocks for the Pulse Output Unit</b>                 | <b>11-1</b> |
| 11.1      | Basic Configuration of Function Blocks for the Pulse Output Unit | 11-2        |
| 11.1.1    | Specifications of the Function Block                             | 11-2        |
| 11.1.2    | Notes for Executing the Function Block                           | 11-3        |
| 11.2      | Function Blocks for the Pulse Output Unit                        | 11-4        |
| 11.2.1    | PG_Power   | 11-4        |
| 11.2.2    | PG_Jog   | 11-5        |
| 11.2.3    | PG_MoveAbsolute  | 11-6        |

|         |                          |       |
|---------|--------------------------|-------|
| 11.2.4  | PG_MoveRelative .....    | 11-7  |
| 11.2.5  | PG_LatchPosition.....    | 11-8  |
| 11.2.6  | PG_Pulser.....           | 11-10 |
| 11.2.7  | PG_Stop.....             | 11-11 |
| 11.2.8  | PG_Home.....             | 11-12 |
| 11.2.9  | PG_SetPosition.....      | 11-13 |
| 11.2.10 | PG_WriteParameter.....   | 11-14 |
| 11.2.11 | PG_ReadParameter.....    | 11-17 |
| 11.2.12 | PG_ClearError.....       | 11-18 |
| 11.2.13 | PG_ReadStatus .....      | 11-18 |
| 11.3    | Error Codes.....         | 11-21 |
| 11.3.1  | Error Check Method ..... | 11-21 |
| 11.3.2  | PG_ERROR .....           | 11-22 |

## **12 Reference Information.....12-1**

|        |   |       |
|--------|---|-------|
| 12.1   | Motion Errors (SMC_ERROR Type).....                       | 12-2  |
| 12.1.1 | Error Check Method.....                                   | 12-2  |
| 12.1.2 | SMC_ERROR.....  | 12-3  |
| 12.2   | RTEX communication error.....                             | 12-11 |
| 12.2.1 | RTEX Error ID.....  | 12-11 |
| 12.2.2 | Alarm Codes .....   | 12-14 |
| 12.2.3 | Warning Codes .....                                       | 12-18 |
| 12.3   | List of AMP Parameters .....                              | 12-21 |
| 12.3.1 | Class 0: Basic Setting .....                              | 12-21 |
| 12.3.2 | Class 1: Gain Adjustment .....                            | 12-21 |
| 12.3.3 | Class 2: Vibration Suppression Function .....             | 12-22 |
| 12.3.4 | Class 3: Speed, Torque Control, Full-closed Control ..... | 12-23 |
| 12.3.5 | Class 4: I/O Monitor Setting .....                        | 12-24 |
| 12.3.6 | Class 5: Enhancing Setting.....                           | 12-25 |
| 12.3.7 | Class 6: Special Setting 1 .....                          | 12-27 |
| 12.3.8 | Class 7: Special Setting 2 .....                          | 12-29 |
| 12.3.9 | Class 8: Special Setting 3.....                           | 12-31 |
| 12.4   | Monitor Commands.....                                     | 12-32 |

# 1 List of Instructions









---

|   |      |
|---|------|
| 1.1 List of Ladder Instructions.....  | 1-2  |
| 1.2 List of Function Instructions .....   | 1-3  |
| 1.3 List of Function Block Instructions .....                                     | 1-9  |
| 1.3.1 Basic Instructions .....  | 1-9  |
| 1.3.2 Motion Control Function Blocks (Single Axis Control) .....                  | 1-11 |
| 1.3.3 Motion Control Function Blocks (Synchronous Control) .....                  | 1-14 |
| 1.3.4 Motion Control Function Blocks (Multi-axis Control) .....                   | 1-15 |
| 1.3.5 Motion Control Function Blocks (Motion Communication Control) ..            | 1-15 |
| 1.3.6 Motion Control Function Blocks (Auxiliary Function) .....                   | 1-18 |
| 1.3.7 Function Blocks (Others) .....  | 1-20 |
| 1.3.8 Function Blocks (For the GM1 Pulse Output Unit).....                        | 1-27 |
| 1.4 List of Function Block Instructions that Cannot Be Used with the<br>GM1 ..... | 1-29 |

## 1.1 List of Ladder Instructions

### 1.1 List of Ladder Instructions

The following table lists contact and coil ladder instructions that can be used in ladder diagram programs for GM Programmer.

| Name                   | Code  | Description  | Simulation (●: Supported, -: Not supported) | Page    |
|------------------------|---|--|---|---------|
| NO contact             |    | This instruction outputs a BOOL-type input from the left to the right.<br>If the variable of the contact is TRUE, then the input value from the left is output.<br>If the variable of the contact is FALSE, then FALSE is output.                        | ●   | "P.2-2" |
| NC contact             |    | This instruction outputs the negated value of the BOOL-type input from the left to the right.<br>If the variable of the contact is TRUE, then FALSE is output.<br>If the variable of the contact is FALSE, then the input value from the left is output. | ●   | "P.2-3" |
| Rising edge detection  |    | When a rising edge is detected in the BOOL-type input from the left, TRUE is output for one cycle only.  | ●   | "P.2-3" |
| Falling edge detection |    | When a falling edge is detected in the BOOL-type input from the left, TRUE is output for one cycle only  | ●   | "P.2-4" |
| Parallel NO contact    | -   | NO contacts can be wired in parallel.<br>The contacts wired in parallel are treated as OR logic. If the output of one or more contacts is TRUE, TRUE is output.  | ●   | "P.2-5" |
| Parallel NC contact    | -   | NC contacts can be wired in parallel.<br>The contacts wired in parallel are treated as OR logic. If the output of one or more contacts is TRUE, TRUE is output.  | ●   | "P.2-6" |
| Coil                   |  | A BOOL-type input from the left can be saved.<br>If the input is TRUE, then TRUE is saved.<br>If the input is FALSE, then FALSE is saved.  | ●   | "P.2-7" |
| Negated coil           |  | The negated value of the BOOL-type input from the left can be saved.<br>If the input is TRUE, then FALSE is saved.<br>If the input is FALSE, then TRUE is saved.   | ●   | "P.2-7" |
| Set coil               |  | If the BOOL-type input from the left becomes TRUE, TRUE is saved.<br>It can be used together with the reset coil.  | ●   | "P.2-8" |
| Reset coil             |  | If the BOOL-type input from the left becomes TRUE, FALSE is saved.<br>It can be used together with the set coil.   | ●   | "P.2-9" |

## 1.2 List of Function Instructions

This section provides lists of the functions used by the GM Programmer. These functions can be used without declaring them with variables.

### ■ Substitution instruction

| Name | Function     | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|------|--------------|---|----------------------------------|----------|---|---------|
|      |              |   | (●: Supported, -: Not supported) |          |   |         |
| MOVE | Substitution | Substitutes the input argument values with the output argument. | ●                                | ●        | ●   | "P.3-3" |

### ■ Arithmetic operation instructions

| Name | Function       | Overview                                     | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|------|----------------|--|----------------------------------|----------|---|---------|
|      |                |  | (●: Supported, -: Not supported) |          |   |         |
| ADD  | Addition       | Adds the input arguments.                    | ●                                | ●        | ●   | "P.3-4" |
| SUB  | Subtraction    | Subtracts the input arguments.               | ●                                | ●        | ●   | "P.3-5" |
| MUL  | Multiplication | Multiplies the input arguments.              | ●                                | ●        | ●   | "P.3-6" |
| DIV  | Division       | Divides the input arguments.                 | ●                                | ●        | ●   | "P.3-7" |
| MOD  | Mod            | Outputs the remainder of the input argument. | ●                                | ●        | ●   | "P.3-8" |

### ■ Boolean operation instructions

| Name     | Function     | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|----------|--------------|--|----------------------------------|----------|---|----------|
|          |              |  | (●: Supported, -: Not supported) |          |   |          |
| AND      | Logical AND  | Outputs the logical AND of the input arguments.  | ●                                | ●        | ●   | "P.3-9"  |
| OR       | Logical OR   | Outputs the logical OR of the input arguments.   | ●                                | ●        | ●   | "P.3-10" |
| XOR      | Exclusive OR | Outputs the Exclusive OR of the input arguments. | ●                                | ●        | ●   | "P.3-11" |
| NOT      | Negation     | Outputs the negation of the input argument.      | ●                                | ●        | ●   | "P.3-10" |
| AND_THEN | Logical AND  | Outputs the logical AND of the input arguments.  | ●                                | ●        | ●   | "P.3-12" |
| OR_ELSE  | Logical OR   | Outputs the logical OR of the input arguments.   | ●                                | ●        | ●   | "P.3-13" |

## 1.2 List of Function Instructions

### ■ Comparison operation instructions

| Name | Function                           | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------|------------------------------------|---|----------------------------------|----------|---|----------|
|      |                                    |   | (●: Supported, -: Not supported) |          |   |          |
| EQ   | "Equal" comparison                 | Compares the two input arguments and, if they are equal to each other, outputs TRUE.                                    | ●                                | ●        | ●   | "P.3-15" |
| NE   | "Not Equal" comparison             | Compares the two input arguments and, if they are not equal to each other, outputs TRUE.                                | ●                                | ●        | ●   | "P.3-15" |
| LT   | "Less Than" comparison             | Compares the two input arguments and, if the first argument is less than the second argument, outputs TRUE.             | ●                                | ●        | ●   | "P.3-16" |
| LE   | "Less Than or Equal" comparison    | Compares the two input arguments and, if the first argument is less than the second argument or equal, outputs TRUE.    | ●                                | ●        | ●   | "P.3-17" |
| GT   | "Greater Than" comparison          | Compares the two input arguments and, if the first argument is greater than the second argument, outputs TRUE.          | ●                                | ●        | ●   | "P.3-18" |
| GE   | "Greater Than Or Equal" comparison | Compares the two input arguments and, if the first argument is greater than the second argument or equal, outputs TRUE. | ●                                | ●        | ●   | "P.3-19" |

### ■ Bit shift instructions

| Name | Function    | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------|-------------|--|----------------------------------|----------|---|----------|
|      |             |  | (●: Supported, -: Not supported) |          |   |          |
| SHL  | Shift left  | Shifts the input argument to the left by the specified number of bits. Inserts "0" from the least significant bit to the specified bit and outputs the data. | ●                                | ●        | ●   | "P.3-21" |
| SHR  | Shift right | Shifts the input argument to the right by the specified number of bits. Inserts "0" from the most significant bit to the specified bit and outputs the data. | ●                                | ●        | ●   | "P.3-22" |
| ROL  | Rotate      | Shifts the input argument to the left by the specified   | ●                                | ●        | ●   | "P.3-22" |



## 1.2 List of Function Instructions

| Name | Function     | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------|--------------|---|----------------------------------|----------|---|----------|
|      |              |   | (●: Supported, -: Not supported) |          |   |          |
|      | left         | number of bits. Inserts the value in excess from the most significant bit into the data starting from the least significant bit and outputs the data.   |                                  |          |   |          |
| ROR  | Rotate right | Shifts the input argument to the right by the specified number of bits. Inserts the value in excess from the least significant bit into the data starting from the most significant bit and outputs the data. | ●                                | ●        | ●   | "P.3-23" |

### ■ Selection instructions

| Name  | Function        | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|-------|-----------------|---|----------------------------------|----------|---|----------|
|       |                 |   | (●: Supported, -: Not supported) |          |   |          |
| SEL   | Binary selector | Outputs "IN0" when the input argument G is FALSE and "IN1" when the input argument G is TRUE.         | ●                                | ●        | ●   | "P.3-25" |
| MUX   | Multiplexer     | Outputs the input argument value depending on the input argument K (0,1,2,...).                       | ●                                | ●        | ●   | "P.3-26" |
| LIMIT | Limiter         | Limits the value of the input argument IN between the input arguments MN and MX and outputs the data. | ●                                | ●        | ●   | "P.3-27" |
| MAX   | Maximum value   | Outputs the maximum value of the input argument.  | ●                                | ●        | ●   | "P.3-28" |
| MIN   | Minimum value   | Outputs the minimum value of the input argument.  | ●                                | ●        | ●   | "P.3-28" |

### ■ Numerical operation instructions

| Name | Function       | Overview                    | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------|----------------|-----------------------------|----------------------------------|----------|---|----------|
|      |                |                             | (●: Supported, -: Not supported) |          |   |          |
| ABS  | Absolute value | Outputs the absolute value. | ●                                | ●        | ●   | "P.3-30" |

## 1.2 List of Function Instructions

| Name | Function                             | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------|--------------------------------------|---|----------------------------------|----------|---|----------|
|      |                                      |   | (●: Supported, -: Not supported) |          |   |          |
| SQRT | Square root                          | Outputs the the square root ( $\sqrt{\phantom{x}}$ ) of a number. | ●                                | ●        | ●   | "P.3-30" |
| LN   | Natural logarithm                    | Outputs the natural logarithm ( $\log_e X$ ) of a number.         | ●                                | ●        | ●   | "P.3-31" |
| LOG  | Common logarithm                     | Outputs the common logarithm ( $\log_{10} X$ ) of a number.       | ●                                | ●        | ●   | "P.3-32" |
| EXP  | Natural exponent                     | Outputs the natural exponent ( $e^X$ ) of a number.               | ●                                | ●        | ●   | "P.3-33" |
| EXPT | Exponentiation                       | Outputs the exponentiation of a number ( $a^n$ ).                 | ●                                | ●        | ●   | "P.3-33" |
| SIN  | Trigonometric function (sine)        | Outputs the result of the sine function calculation.              | ●                                | ●        | ●   | "P.3-34" |
| COS  | Trigonometric function (cosine)      | Outputs the result of the cosine function calculation.            | ●                                | ●        | ●   | "P.3-35" |
| TAN  | Trigonometric function (tangent)     | Outputs the result of the tangent function calculation.           | ●                                | ●        | ●   | "P.3-36" |
| ASIN | Trigonometric function (arc sine)    | Outputs the result of the arc sine function calculation.          | ●                                | ●        | ●   | "P.3-37" |
| ACOS | Trigonometric function (arc cosine)  | Outputs the result of the arc cosine function calculation.        | ●                                | ●        | ●   | "P.3-38" |
| ATAN | Trigonometric function (arc tangent) | Outputs the result of the arc tangent function calculation.       | ●                                | ●        | ●   | "P.3-38" |

### ■ Data type conversion instructions

| Name                 | Function             | Overview                                       | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|----------------------|----------------------|--|----------------------------------|----------|---|----------|
|                      |                      |  | (●: Supported, -: Not supported) |          |   |          |
| <Type 1>_TO_<Type 2> | Data type conversion | Converts type 1 input argument to type 2.      | ●                                | ●        | ●   | "P.3-40" |
| TRUNC                | Data type conversion | Changes the real number to the DINT-type data. | ●                                | ●        | ●   | "P.3-41" |
| TRUNC_INT            | Data type conversion | Changes the real number to the INT-type data.  | ●                                | ●        | ●   | "P.3-41" |

### ■ Character string instructions

The standard library is required.

| Name    | Function  | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|---------|---|---|----------------------------------|----------|---|----------|
|         |   |   | (●: Supported, -: Not supported) |          |   |          |
| LEN     | Length of a character string                      | Outputs the length of a character string.   | ●                                | ●        | ●   | "P.3-43" |
| LEFT    | Extracting characters from the left end           | Extracts a character string consisting of the specified number of characters from the left of the character string.   | ●                                | ●        | ●   | "P.3-43" |
| RIGHT   | Extracting characters from the right end          | Extracts a character string consisting of the specified number of characters from the right of the character string.  | ●                                | ●        | ●   | "P.3-44" |
| MID     | Extracting characters from the specified position | Extracts a character string consisting of the specified number of characters from the specified position of the character string.                                 | ●                                | ●        | ●   | "P.3-45" |
| CONCAT  | Concatenating character strings                   | Concatenates two character strings.   | ●                                | ●        | ●   | "P.3-46" |
| INSERT  | Inserting a character string                      | Inserts another character string into the specified position of one character string.   | ●                                | ●        | ●   | "P.3-47" |
| DELETE  | Deleting a character string                       | Deletes a character string consisting of the specified number of characters from the specified position of the character string.                                  | ●                                | ●        | ●   | "P.3-48" |
| REPLACE | Replacing a character string                      | Replaces a character string, consisting of the specified number of characters from the specified position of the character string, with another character string. | ●                                | ●        | ●   | "P.3-49" |
| FIND    | Search for a character string                     | Searches for a specified character string in the character strings and outputs the position.  | ●                                | ●        | ●   | "P.3-50" |

## 1.2 List of Function Instructions

---

### ■ Other instructions

| Name   | Function        | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|--------|-----------------|--|----------------------------------|----------|---|----------|
|        |                 |  | (●: Supported, -: Not supported) |          |   |          |
| SIZEOF | Get the size    | Outputs the size (in units of byte) of the input argument. | ●                                | ●        | ●   | "P.3-52" |
| ADR    | Get the address | Outputs the address of the input argument.                 | ●                                | ●        | ●   | "P.3-52" |

### ■ SD memory card slot instruction

| Name                | Function                             | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|---------------------|--------------------------------------|---|----------------------------------|----------|---|----------|
|                     |                                      |   | (●: Supported, -: Not supported) |          |   |          |
| SYS_GetSDCoverState | Get SD card cover open / close state | Gets an open / close state of the card cover for the SD memory card slot. | ●                                | ●        | -   | "P.3-54" |
| SYS_GetSDAccessRdy  | Get SD card access ready state       | Gets the state whether an access to the SD memory card is allowed.        | ●                                | ●        | -   | "P.3-54" |

## 1.3 List of Function Block Instructions

This section provides lists of the function blocks used by the GM Programmer. These function blocks can be used with declaring the instances with variables.

### 1.3.1 Basic Instructions

#### ■ Timer instructions

| Name | Function       | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|------|----------------|--|----------------------------------|----------|---|---------|
|      |                |  | (●: Supported, -: Not supported) |          |   |         |
| TON  | Timer ON       | Starts the timer when the input argument changes from FALSE to TRUE and, after an elapse of the specified time, the output argument outputs TRUE.                                  | ●                                | ●        | ●   | "P.4-2" |
| TOF  | Timer OFF      | Starts the timer when the input argument changes from TRUE to FALSE and, after an elapse of the specified time, the output argument outputs FALSE.                                 | ●                                | ●        | ●   | "P.4-3" |
| TP   | Timer pulse    | Starts the timer when the input argument changes from FALSE to TRUE until the specified time elapses. Outputs TRUE to the output argument while the timer keeps counting.          | ●                                | ●        | ●   | "P.4-4" |
| RTC  | Realtime clock | Starts counting time from the specified date and time when the input argument changes from FALSE to TRUE. Outputs TRUE to the output argument while the clock keeps counting time. | ●                                | ●        | ●   | "P.4-6" |

#### ■ Counter instructions

| Name | Function   | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|------|------------|---|----------------------------------|----------|---|---------|
|      |            |   | (●: Supported, -: Not supported) |          |   |         |
| CTU  | Up counter | Starts incrementing the counter value at the rising edge of the input argument CU and, after counting the specified number of count values, outputs TRUE. | ●                                | ●        | ●   | "P.4-7" |

## 1.3 List of Function Block Instructions

| Name | Function        | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|------|-----------------|--|----------------------------------|----------|---|---------|
|      |                 |  | (●: Supported, -: Not supported) |          |   |         |
| CTD  | Down counter    | Starts decrementing from the specified number of count value at the rising edge of the input argument CD. Outputs TRUE when it reaches 0.  | ●                                | ●        | ●   | "P.4-8" |
| CTUD | Up-down counter | Starts incrementing the counter value at the rising edge of the input argument CU and, after counting the specified number of count values, outputs TRUE.<br><br>Starts decrementing the counter value at the rising edge of the input argument CD and, when it reaches 0, outputs TRUE. | ●                                | ●        | ●   | "P.4-9" |

### ■ Edge detection instructions

| Name   | Function               | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|--------|------------------------|--|----------------------------------|----------|---|----------|
|        |                        |  | (●: Supported, -: Not supported) |          |   |          |
| R_TRIG | Rising edge detection  | Outputs TRUE for one cycle only when detecting a rising edge.  | ●                                | ●        | ●   | "P.4-11" |
| F_TRIG | Falling edge detection | Outputs TRUE for one cycle only when detecting a falling edge. | ●                                | ●        | ●   | "P.4-11" |

### ■ Bistable circuit instructions

| Name | Function                        | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------|---------------------------------|---|----------------------------------|----------|---|----------|
|      |                                 |   | (●: Supported, -: Not supported) |          |   |          |
| SR   | Set-priority bistable circuit   | If the input argument SET1 is TRUE, outputs TRUE.<br><br>If the input argument RESET is TRUE, outputs FALSE.<br><br>If both SET1 and RESET are TRUE, outputs TRUE | ●                                | ●        | ●   | "P.4-13" |
| RS   | Reset-priority bistable circuit | If the input argument SET1 is TRUE, outputs TRUE.   | ●                                | ●        | ●   | "P.4-14" |

## 1.3 List of Function Block Instructions

| Name | Function | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page |
|------|----------|--|----------------------------------|----------|---|------|
|      |          |  | (●: Supported, -: Not supported) |          |   |      |
|      |          | <p>If the input argument RESET is TRUE, outputs FALSE.</p> <p>If both SET1 and RESET1 are TRUE, outputs FALSE.</p> |                                  |          |   |      |

### 1.3.2 Motion Control Function Blocks (Single Axis Control)

#### ■ Servo ON

| Name     | Function | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|----------|----------|--|----------------------------------|----------|---|---------|
|          |          |  | (●: Supported, -: Not supported) |          |   |         |
| MC_Power | Servo ON | Sets the axis to the servo ON state to be ready for operation. | ●                                | ●        | ●   | "P.5-2" |

#### ■ Home return

| Name     | Function    | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|----------|-------------|---|----------------------------------|----------|---|---------|
|          |             |   | (●: Supported, -: Not supported) |          |   |         |
| PMC_Home | Home return | Performs home return operation on the axis. Uses the home return function of the amplifier. | ●                                | -        | -   | "P.5-4" |
| MC_Home  | Home return | Performs home return operation on the axis. Uses the home return function of the amplifier. | -                                | ●        | ●   | "P.5-6" |

#### ■ Control switch

| Name                  | Function           | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|-----------------------|--------------------|--|----------------------------------|----------|---|---------|
|                       |                    |  | (●: Supported, -: Not supported) |          |   |         |
| SMC_SetControllerMode | Control mode setup | Sets up the control mode for controlling the position, velocity, and torque. | ●                                | ●        | -   | "P.5-8" |

## 1.3 List of Function Block Instructions

### ■ Stop

| Name    | Function    | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|---------|-------------|---|----------------------------------|----------|---|----------|
|         |             |   | (●: Supported, -: Not supported) |          |   |          |
| MC_Stop | Forced stop | Causes the axis to make a deceleration stop. After stopping, the axis remains stopped while Execute is TRUE.  | ●                                | ●        | ●   | "P.5-9"  |
| MC_Halt | Stop        | Causes the axis to make a deceleration stop. After the axis is stopped or while the axis is being decelerated, other motion instructions can be executed. | ●                                | ●        | ●   | "P.5-10" |

### ■ JOG / Inching

| Name     | Function | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|----------|----------|--|----------------------------------|----------|---|----------|
|          |          |  | (●: Supported, -: Not supported) |          |   |          |
| MC_Jog   | Jogging  | Causes the axis to keep traveling in a forward or reverse direction at a constant velocity while the input is TRUE.      | ●                                | ●        | ●   | "P.5-11" |
| SMC_Inch | Inching  | Causes the axis to travel in a forward or reverse direction for a specified relative distance when the input turns TRUE. | ●                                | ●        | ●   | "P.5-12" |

### ■ Position control

| Name                | Function                   | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|---------------------|----------------------------|---|----------------------------------|----------|---|----------|
|                     |                            |   | (●: Supported, -: Not supported) |          |   |          |
| MC_MoveAbsolute     | Absolute value positioning | Causes the axis to travel to a position specified as an absolute position.                | ●                                | ●        | ●   | "P.5-14" |
| MC_MoveRelative     | Relative value positioning | Causes the axis to travel to a position specified as a relative position.                 | ●                                | ●        | ●   | "P.5-15" |
| MC_MoveAdditive     | Change target position     | Adds a relative distance to the target position of the immediately preceding instruction. | ●                                | ●        | ●   | "P.5-16" |
| MC_MoveSuperImposed | Superimposed positioning   | Adds a relative distance, a velocity, an acceleration, and a deceleration to the          | ●                                | ●        | ●   | "P.5-18" |



## 1.3 List of Function Block Instructions

| Name                       | Function                              | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|----------------------------|---------------------------------------|--|----------------------------------|----------|---|----------|
|                            |                                       |  | (●: Supported, -: Not supported) |          |   |          |
|                            |                                       | operations of the immediately preceding instruction.   |                                  |          |   |          |
| MC_PositionProfile         | Position profile move                 | Causes the axis to operate according to the profile data that consists of a combination of position and time.                                | ●                                | ●        | ●   | "P.5-20" |
| SMC_MoveContinuousAbsolute | Absolute value position velocity move | Executes absolute value positioning and, after the axis reaches the target position, causes the axis to keep moving at a specified velocity. | ●                                | ●        | ●   | "P.5-24" |
| SMC_MoveContinuousRelative | Relative value position velocity move | Executes relative value positioning and, after the axis reaches the target position, causes the axis to keep moving at a specified velocity. | ●                                | ●        | ●   | "P.5-25" |

### ■ Velocity control

| Name                   | Function                  | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------------------------|---------------------------|---|----------------------------------|----------|---|----------|
|                        |                           |   | (●: Supported, -: Not supported) |          |   |          |
| MC_MoveVelocity        | Velocity control          | Specifies the velocity of the axis.   | ●                                | ●        | ●   | "P.5-27" |
| MC_VelocityProfile     | Velocity profile move     | Causes the axis to operate according to the profile data that consists of a combination of time and velocity.     | ●                                | ●        | ●   | "P.5-28" |
| MC_AccelerationProfile | Acceleration profile move | Causes the axis to operate according to the profile data that consists of a combination of time and acceleration. | ●                                | ●        | ●   | "P.5-29" |

### ■ Torque control

| Name          | Function       | Overview                                | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|---------------|----------------|---|----------------------------------|----------|---|----------|
|               |                |   | (●: Supported, -: Not supported) |          |   |          |
| PMC_SetTorque | Torque control | Specifies by % the torque of the axis.  | ●                                | ●        | ●   | "P.5-32" |
| SMC_SetTorque | Torque control | Specifies by Nm the torque of the axis. | -                                | ●        | ●   | "P.5-33" |

## 1.3 List of Function Block Instructions

### 1.3.3 Motion Control Function Blocks (Synchronous Control)

#### ■ Cam operation

| Name               | Function             | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|--------------------|----------------------|--|----------------------------------|----------|---|---------|
|                    |                      |  | (●: Supported, -: Not supported) |          |   |         |
| MC_CamIn           | Start cam control    | Starts cam synchronous operation.                      | ●                                | ●        | ●   | "P.6-2" |
| MC_CamOut          | Cancel cam operation | Cancels cam synchronous operation.                     | ●                                | ●        | ●   | "P.6-5" |
| MC_CamTable Select | Cam table selection  | Specifies the cam table for cam synchronous operation. | ●                                | ●        | ●   | "P.6-6" |
| SMC_GetTappetValue | Tappet output        | Outputs the tappet set in the cam table.               | ●                                | ●        | ●   | "P.6-8" |

#### ■ Gear operation

| Name         | Function                          | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|--------------|-----------------------------------|---|----------------------------------|----------|---|----------|
|              |                                   |   | (●: Supported, -: Not supported) |          |   |          |
| MC_GearIn    | Start gear operation              | Starts gear synchronous operation.                                      | ●                                | ●        | ●   | "P.6-11" |
| MC_GearInPos | Position specified gear operation | Starts gear synchronous operation from the specified absolute position. | ●                                | ●        | ●   | "P.6-12" |
| MC_GearOut   | Cancel gear operation             | Cancels the gear synchronous operation.                                 | ●                                | ●        | ●   | "P.6-14" |

#### ■ Phase correction

| Name       | Function                     | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------------|------------------------------|---|----------------------------------|----------|---|----------|
|            |                              |   | (●: Supported, -: Not supported) |          |   |          |
| MC_Phasing | Master axis phase correction | Corrects the phase between the master and slave axes. | ●                                | ●        | ●   | "P.6-16" |

### 1.3.4 Motion Control Function Blocks (Multi-axis Control)

#### ■ Interpolation control

| Name               | Function                     | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page    |
|--------------------|------------------------------|--|----------------------------------|----------|---|---------|
|                    |                              |  | (●: Supported, -: Not supported) |          |   |         |
| PMC_Interpolator2D | 2-axis interpolation control | Specifies the CNC pattern and performs 2-axis interpolation control. | ●                                | ●        | ●   | "P.7-2" |
| PMC_Interpolator3D | 3-axis interpolation control | Specifies the CNC pattern and performs 3-axis interpolation control. | ●                                | ●        | ●   | "P.7-3" |
| PMC_NCDecoder      | CNC table conversion         | Converts a CNC table into executable format.                         | ●                                | ●        | ●   | "P.7-5" |

### 1.3.5 Motion Control Function Blocks (Motion Communication Control)

#### ■ RTEX

| Name                               | Function                          | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------------------------------------|-----------------------------------|---|----------------------------------|----------|---|----------|
|                                    |                                   |   | (●: Supported, -: Not supported) |          |   |          |
| RTEX_ClearAlarm                    | Clear amplifier alarm             | Clears the amplifier's alarm.                       | ●                                | -        | -   | "P.8-2"  |
| RTEX_ReadAlarm                     | Read amplifier alarm              | Reads the amplifier's alarm.                        | ●                                | -        | -   | "P.8-4"  |
| RTEX_ReadAlarmState                | Amplifier alarm status            | Reads the amplifier's alarm status.                 | ●                                | -        | -   | "P.8-5"  |
| RTEX_ReadAlarmData                 | Amplifier monitor                 | Reads the amplifier's monitor data.                 | ●                                | -        | -   | "P.8-6"  |
| RTEX_ReadAlarmParameter            | Read amplifier parameter          | Reads the amplifier's parameters.                   | ●                                | -        | -   | "P.8-7"  |
| RTEX_WriteAlarmParameter           | Write amplifier parameter         | Writes the amplifier's parameters.                  | ●                                | -        | -   | "P.8-8"  |
| RTEX_WriteAlarmEEPROM              | Write amplifier EEPROM            | Writes parameters of the servo amplifier to EEPROM. | ●                                | -        | -   | "P.8-9"  |
| RTEX_Reset                         | Reset RTEX                        | Resets the entire RTEX network.                     | ●                                | -        | -   | "P.8-9"  |
| RTEX_ClearAlarmMultiTurnData       | Clear the multi-turn data         | Clears the multi-turn data of the amplifier.        | ●                                | -        | -   | "P.8-10" |
| RTEX_ClearAlarmPositionalDeviation | Clear amplifier deviation counter | Clears the deviation counter of the amplifier.      | ●                                | -        | -   | "P.8-11" |

## 1.3 List of Function Block Instructions

| Name                         | Function              | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------------------------------|-----------------------|---|----------------------------------|----------|---|----------|
|                              |                       |   | (●: Supported, -: Not supported) |          |   |          |
| RTEX_GetTrackingCommandError | Error                 | Measures the number of sent RTEX commands and the number of lost RTEX commands. | ●                                | -        | -   | "P.8-12" |
| RTEX_ReadPot                 | Read NOT of amplifier | Reads the amplifier's NOT status.   | ●                                | -        | -   | "P.8-13" |
| RTEX_ReadPot                 | Read POT of amplifier | Reads the amplifier's POT status.   | ●                                | -        | -   | "P.8-14" |

### ■ EtherCAT

| Name                   | Function                                       | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|------------------------|--|--|----------------------------------|----------|---|----------|
|                        |  |  | (●: Supported, -: Not supported) |          |   |          |
| ETC_CO_SdoRead         | Read slave parameter                           | Reads EtherCAT slave parameters.<br>Unlike ETC_CO_SdoRead4, this FB supports parameters longer than 4 bytes.   | -                                | ●        | -   | "P.8-15" |
| ETC_CO_SdoRead4        | Read four bytes of slave parameter             | Reads EtherCAT slave parameters.<br>Unlike ETC_CO_SdoRead, this FB supports only parameters with 4 bytes or less.  | -                                | ●        | -   | "P.8-16" |
| ETC_CO_SdoReadDWord    | Read double words of slave parameter           | Just like ETC_CO_SdoRead4, this FB reads the EtherCAT slave parameters.<br>The read data is stored in DWORD (dwData), not in an array.   | -                                | ●        | -   | "P.8-17" |
| ETC_CO_SdoRead_Access  | Read slave parameter index                     | Just like ETC_CO_SdoRead, this FB reads the EtherCAT slave parameters.<br>By setting the xCompleteAccess input to TRUE and the bySubIndex input to 0, you can read complete indexes including all entries. | -                                | ●        | -   | "P.8-18" |
| ETC_CO_SdoRead_Channel | Read priority specification of slave parameter | Reads all EtherCAT slave parameters.   | -                                | ●        | -   | "P.8-19" |

## 1.3 List of Function Block Instructions

| Name                             | Function                              | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|----------------------------------|---------------------------------------|---|----------------------------------|----------|---|----------|
|                                  |                                       |   | (●: Supported, -: Not supported) |          |   |          |
| ETC_CO_SdoWrite                  | Write slave parameter                 | Writes EtherCAT slave parameters.<br>Unlike ETC_CO_SdoWrite4, this FB supports parameters longer than 4 bytes.  | -                                | ●        | -   | "P.8-21" |
| ETC_CO_SdoWrite4                 | Write four bytes of slave parameter   | Writes EtherCAT slave parameters.<br>Unlike ETC_CO_SdoWrite, this FB supports only parameters with 4 bytes or less.   | -                                | ●        | -   | "P.8-22" |
| ETC_CO_SdoWriteDWord             | Write double words of slave parameter | Just like ETC_CO_SdoWrite4, this FB writes the EtherCAT slave parameters.<br>The write data is transferred in DWORD (dwData), not in an array.  | -                                | ●        | -   | "P.8-24" |
| ETC_CO_SdoWrite_Access           | Write slave parameter index           | Just like ETC_CO_SdoWrite, this FB writes the EtherCAT slave parameters.<br>By setting the xCompleteAccess input to TRUE and the bySubIndex input to 0, you can write complete indexes including all entries.<br>By using the byChannelPriority (BYTE) input, you can specify the channel and priority using a CoE mailbox message. | -                                | ●        | -   | "P.8-25" |
| ReadIdentification               | Read slave identification data        | Reads identification data from EtherCAT slaves.   | -                                | ●        | -   | "P.8-26" |
| ReadMemory                       | Read slave memory                     | Reads the EtherCAT slave memory.  | -                                | ●        | -   | "P.8-27" |
| ReadNbrSlaves                    | Read the number of connected slaves   | Reads the number of slaves currently connected.   | -                                | ●        | -   | "P.8-28" |
| WriteMemory                      | Write slave memory                    | Writes the EtherCAT slave memory.   | -                                | ●        | -   | "P.8-29" |
| PETC_ClearAmpPositionalDeviation | Clear amplifier deviation counter     | Clears the deviation counter of the amplifier.  | -                                | ●        | -   | "P.8-30" |

## 1.3 List of Function Block Instructions

| Name                            | Function                                  | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|---------------------------------|---|---|----------------------------------|----------|---|----------|
|                                 |   |   | (●: Supported, -: Not supported) |          |   |          |
| IoDrvEtherCAT<br>.GetStatistics | Get EtherCAT frame statistics information | Gets the EtherCAT frame statistics information. | -                                | ●        | -   | "P.8-31" |

### 1.3.6 Motion Control Function Blocks (Auxiliary Function)

#### ■ Motion auxiliary function (Monitoring)

| Name                       | Function                       | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|----------------------------|--------------------------------|---|----------------------------------|----------|---|----------|
|                            |                                |   | (●: Supported, -: Not supported) |          |   |          |
| MC_ReadActualPosition      | Read actual position           | Reads the actual position data of the axis.   | ●                                | ●        | ●   | "P.9-2"  |
| MC_ReadActualVelocity      | Read actual velocity           | Reads the actual velocity of the axis.  | ●                                | ●        | ●   | "P.9-2"  |
| PMC_ReadActualTorque       | Read actual torque             | Read the actual torque value of the axis.   | ●                                | ●        | -   | "P.9-3"  |
| MC_ReadActualTorque        | Read actual torque             | Reads the actual torque value of the axis.  | -                                | ●        | -   | "P.9-4"  |
| MC_ReadAxisError           | Read axis error                | Gets general axis errors not related to function blocks.  | -                                | ●        | ●   | "P.9-4"  |
| MC_ReadStatus              | Read status                    | Reads the status information of the axis.   | ●                                | ●        | ●   | "P.9-5"  |
| SMC_InPosition             | In-position judgment           | Compares the actual position of the AMP with the command value and judges whether the position is within the specified range. | ●                                | ●        | ●   | "P.9-7"  |
| SMC_ReadFBError            | Read oldest error              | Reads the oldest function block error information.  | ●                                | ●        | ●   | "P.9-8"  |
| SMC_ClearFBError           | Clear oldest error             | Clears the oldest FB error information.   | ●                                | ●        | ●   | "P.9-9"  |
| SMC_CheckAxisCommunication | Check axis communication state | Checks the communication state of the axis.   | ●                                | ●        | ●   | "P.9-10" |
| SMC_CheckLimits            | Check exceeding limits         | Checks whether the velocity, acceleration, or deceleration is in excess of the dynamic limit set value of the device.         | ●                                | ●        | ●   | "P.9-11" |
| SMC_GetMaxSetAccDec        | Measure maximum                | Measures the maximum value of the axis  | ●                                | ●        | ●   | "P.9-12" |

## 1.3 List of Function Block Instructions

| Name                  | Function                            | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|-----------------------|-------------------------------------|---|----------------------------------|----------|---|----------|
|                       |                                     |   | (●: Supported, -: Not supported) |          |   |          |
|                       | acceleration / deceleration         | acceleration / deceleration command.  |                                  |          |   |          |
| SMC_GetMaxSetVelocity | Measure maximum velocity            | Measures the maximum value of the axis velocity command.                          | ●                                | ●        | ●   | "P.9-12" |
| SMC_GetTrackingError  | Measure tracking error              | Measures the tracking error of the actual position for the axis command position. | ●                                | ●        | ●   | "P.9-13" |
| SMC_MeasureDistance   | Measures turnaround travel distance | Measures the travel distance.   | ●                                | ●        | ●   | "P.9-14" |
| SMC_ReadSetPosition   | Read axis set position              | Reads the set command position of the axis.                                       | ●                                | ●        | ●   | "P.9-15" |

### ■ Motion auxiliary function (Change / reset)

| Name              | Function               | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|-------------------|------------------------|--|----------------------------------|----------|---|----------|
|                   |                        |  | (●: Supported, -: Not supported) |          |   |          |
| MC_Reset          | Reset axis error       | Resets the state transition error of the axis.   | ●                                | ●        | ●   | "P.9-16" |
| SMC3_Reinitialize | Reinitialize axis      | Restarts the servo amplifier / axis.             | -                                | ●        | ●   | "P.9-16" |
| MC_SetPosition    | Change actual position | Changes the actual command position of the axis. | ●                                | ●        | ●   | "P.9-17" |

### ■ Motion auxiliary function (Other functions)

| Name                    | Function                     | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|-------------------------|------------------------------|---|----------------------------------|----------|---|----------|
|                         |                              |   | (●: Supported, -: Not supported) |          |   |          |
| PMC_ReadLatchedPosition | Monitor AMP latch position   | Monitors the AMP latch position.                      | ●                                | -        | -   | "P.9-18" |
| PMC_StopLatchedPosition | Stop AMP latch monitoring    | Stops the axis at the AMP latch position.             | ●                                | -        | -   | "P.9-20" |
| MC_TouchProbe           | Enable AMP latch monitoring  | Reads the axis position when a trigger signal occurs. | -                                | ●        | ●   | "P.9-23" |
| MC_AbortTrigger         | Disable AMP latch monitoring | Aborts the trigger event (MC_TouchProbe).             | -                                | ●        | ●   | "P.9-25" |
| MC_DigitalCamSwitch     | Enable digital cam switch    | Performs ON / OFF control on the digital              | ●                                | ●        | ●   | "P.9-26" |

## 1.3 List of Function Block Instructions

| Name                     | Function            | Overview                               | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|--------------------------|---------------------|--|----------------------------------|----------|---|----------|
|                          |                     |  | (●: Supported, -: Not supported) |          |   |          |
|                          |                     | output according to the axis position. |                                  |          |   |          |
| SMC_BacklashCompensation | Compensate backlash | Compensates the backlash.              | ●                                | ●        | ●   | "P.9-29" |

### 1.3.7 Function Blocks (Others)

#### ■ COM port (General-purpose communication)

The following table lists the function blocks that are used to perform general-purpose communication with the COM port.

| Name      | Function       | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page     |
|-----------|----------------|---|----------------------------------|----------|---|----------|
|           |                |   | (●: Supported, -: Not supported) |          |   |          |
| COM.Open  | Open COM port  | Opens the COM port.   | ●                                | ●        | -   | "P.10-4" |
| COM.Close | Close COM port | Closes the COM port.  | ●                                | ●        | -   | "P.10-6" |
| COM.Read  | Read COM port  | Reads data from the COM port.   | ●                                | ●        | -   | "P.10-7" |
| COM.Write | Write COM port | Writes data to the COM port.  | ●                                | ●        | -   | "P.10-8" |
| COM.ERROR | Error ID       | This is an enumeration type error ID that is output when the COM port (general-purpose communication) function block is executed. | ●                                | ●        | -   | "P.10-8" |

#### ■ COM port (Modbus COM)

The following table lists the instructions that are used to perform ModbusRTU communication with the COM port.

| Name               | Function             | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|--------------------|----------------------|---|----------------------------------|----------|---|-----------|
|                    |                      |   | (●: Supported, -: Not supported) |          |   |           |
| IoDrvModbusComPort | ModbusComPort device | This is a function block that controls the Modbus_Master_COM_Port device. | ●                                | ●        | -   | "P.10-10" |



## 1.3 List of Function Block Instructions

| Name                           | Function                     | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|--------------------------------|------------------------------|--|----------------------------------|----------|---|-----------|
|                                |                              |  | (●: Supported, -: Not supported) |          |   |           |
| IoDrvModbus.ModbusChannel      | Start sending Modbus command | Sends the command set in the Modbus Slave channel of the ModbusSlaveCOM_Port device.   | ●                                | ●        | -   | "P.10-10" |
| IoDrvModbus.ModbusRequest      | Modbus request               | Processes the Modbus command specified by I/O without using the ModbusMasterComPort device.  | ●                                | ●        | -   | "P.10-11" |
| IoDrvModbus.ModbusRequest2     | Modbus request 2             | Like the ModbusRequest, processes the Modbus command specified by I/O without using the ModbusMasterComPort device.                                    | ●                                | ●        | -   | "P.10-12" |
| IoDrvModbus.ModbusSlaveComPort | ModbusSlave ComPort device   | This is a function block that controls the Modbus_Slave_COM_Port device.   | ●                                | ●        | -   | "P.10-14" |
| IoDrvModbus.MB_ErrorCodes      | Error code                   | This is an enumeration type error code that is output when the function block for Modbus communication instruction that uses the COM port is executed. | ●                                | ●        | -   | "P.10-14" |

### ■ LAN port (IoDrvEthernet)

The following table lists the library functions that are used for the network interface to perform communication with the LAN port.

| Name                              | Function                                 | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|-----------------------------------|--|---|----------------------------------|----------|---|-----------|
|                                   |  |   | (●: Supported, -: Not supported) |          |   |           |
| IoDrvEthernet                     | Ethernet device                          | This is a function block that acquires the status of the LANPort device.              | ●                                | ●        | -   | "P.10-16" |
| IoDrvEthernet.IPARRAY_TO_INADDR   | From array type to union type            | This is a function that converts an array type IP address to an INADDR (union type).  | ●                                | ●        | ●   | "P.10-16" |
| IoDrvEthernet.IPARRAY_TO_IPSTRING | From array type to character string type | This is a function that converts an array type IP address to a character string type. | ●                                | ●        | ●   | "P.10-17" |
| IoDrvEthernet.IPARRAY_TO_UDINT    | From array type to UDINT type            | This is a function that converts an array type IP address to a UDINT type.            | ●                                | ●        | ●   | "P.10-17" |

## 1.3 List of Function Block Instructions

| Name                            | Function                                 | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|---------------------------------|--|--|----------------------------------|----------|---|-----------|
|                                 |  |  | (●: Supported, -: Not supported) |          |   |           |
| IoDrvEthernet.IPSTRING_TO_UDINT | From character string type to UDINT type | This is a function that converts a character string type IP address to a UDINT type. | ●                                | ●        | ●   | "P.10-18" |
| IoDrvEthernet.UDINT_TO_IPARRAY  | From UDINT type to array type            | This is a function that converts a UDINT type IP address to an array type.           | ●                                | ●        | ●   | "P.10-18" |
| IoDrvEthernet.UDINT_TO_IPSTRING | From UDINT type to character string type | This is a function that converts a UDINT type IP address to an array type.           | ●                                | ●        | ●   | "P.10-18" |

### ■ LAN port (General-purpose communication)

The following table lists the library functions that are used to perform general-purpose communication with the LAN port using the TCP or UDP protocol.

| Name               | Function              | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|--------------------|-----------------------|--|----------------------------------|----------|---|-----------|
|                    |                       |  | (●: Supported, -: Not supported) |          |   |           |
| NBS.TCP_Client     | Connect to TCP client | Connects to the TCP/IP client.   | ●                                | ●        | -   | "P.10-20" |
| NBS.TCP_Connection | Connect TCP           | Establishes the connection of the client connecting to the connection port opened by TCP_Server. | ●                                | ●        | -   | "P.10-20" |
| NBS.TCP_Read       | Receive TCP data      | Acquires data received by the connection port that is established by TCP_Connection.             | ●                                | ●        | -   | "P.10-21" |
| NBS.TCP_Server     | Connect TCP server    | Opens the specified port as a TCP/IP connection port.  | ●                                | ●        | -   | "P.10-22" |
| NBS.TCP_Write      | Send TCP data         | Sends data to the connection port that is established by TCP_Connection.                         | ●                                | ●        | -   | "P.10-23" |
| NBS.UDP_Peer       | Open UDP port         | Opens the UDP/IP port.   | ●                                | ●        | -   | "P.10-23" |
| NBS.UDP_Receive    | Receive UDP data      | Receives data to the connection handle acquired by UDP_Peer.                                     | ●                                | ●        | -   | "P.10-24" |
| NBS.UDP_Send       | Send UDP data         | Sends data to the connection handle acquired by UDP_Peer.  | ●                                | ●        | -   | "P.10-26" |
| NBS.ERROR          | Error code            | This is an enumeration type error code that is output when the function                          | ●                                | ●        | -   | "P.10-25" |

## 1.3 List of Function Block Instructions

| Name | Function | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page |
|------|----------|---|----------------------------------|----------|---|------|
|      |          |   | (●: Supported, -: Not supported) |          |   |      |
|      |          | block for communication instruction that uses the LAN port is executed. |                                  |          |   |      |

### ■ LAN port (Modbus TCP)

The following table lists the library functions that are used to perform ModbusTCP communication with the LAN port.

| Name                         | Function                     | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|------------------------------|------------------------------|--|----------------------------------|----------|---|-----------|
|                              |                              |  | (●: Supported, -: Not supported) |          |   |           |
| IoDrvModbusTCP               | ModbusTCP device             | This is a function block that controls the Modbus_TCP_Master device.   | ●                                | ●        | -   | "P.10-28" |
| IoDrvModbusTCP.ModbusChannel | Start sending Modbus command | Sends the command set in the Modbus Slave channel of the ModbusTCP_Slave device.   | ●                                | ●        | -   | "P.10-28" |
| IoDrvModbusTCP.ModbusRequest | Modbus request               | Processes the Modbus command specified by I/O without using the Modbus_TCP_Slave device.   | ●                                | ●        | -   | "P.10-29" |
| IoDrvModbusTCP.Slave         | ModbusTCP Slave device       | This is a function block that controls the Modbus_TCP_Slave device.  | ●                                | ●        | -   | "P.10-30" |
| IoDrvModbus.MB_ErrorCodes    | Error code                   | This is an enumeration type error code that is output when the function block for Modbus communication instruction that uses the LAN port is executed. | ●                                | ●        | -   | "P.10-31" |

### ■ LAN port (EtherNet/IP)

The following table lists instructions that are used to control EtherNet/IP scanner and adapter functions using the GM1 controller.

| Name             | Function                   | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|------------------|----------------------------|--|----------------------------------|----------|---|-----------|
|                  |                            |  | (●: Supported, -: Not supported) |          |   |           |
| IoDrvEtherNet/IP | EtherNet/IP scanner device | This is a function block that controls the EtherNet/IP scanner device. | ●                                | ●        | -   | "P.10-33" |

## 1.3 List of Function Block Instructions

| Name                  | Function   | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|-----------------------|--|---|----------------------------------|----------|---|-----------|
|                       |  |   | (●: Supported, -: Not supported) |          |   |           |
| RemoteAdapter         | Remote adapter device                              | This is a function block for the Remote adapter device linked to the EtherNet/IP scanner device.    | ●                                | ●        | -   | "P.10-34" |
| IoDrvEtherNetPAdapter | EtherNet/IP adapter device                         | This is a function block that controls the EtherNet/IP adapter device.                              | ●                                | ●        | -   | "P.10-36" |
| Module                | EtherNet/IP module device                          | This is a function block that controls the EtherNet/IP module device.                               | ●                                | ●        | -   | "P.10-37" |
| Apply_Attributes      | Apply_Attributes service                           | This is a function block that calls Apply_Attributes service of the CIP object instance.            | ●                                | ●        | -   | "P.10-38" |
| Generic_Service       | Execute generic service                            | This is a function block that executes generic services with the EtherNet/IP adapter.               | ●                                | ●        | -   | "P.10-39" |
| Get_Attribute_Single  | Inquire specific attributes of a specific instance | This is a function block that inquires specific attributes of a specific instance of the CIP object | ●                                | ●        | -   | "P.10-40" |
| Get_Attributes_All    | Inquire all attributes of a specific instance      | This is a function block that inquires all attributes of a specific instance of the CIP object      | ●                                | ●        | -   | "P.10-41" |
| Set_Attribute_Single  | Set specific attributes of a specific instance     | This is a function block that sets specific attributes of a specific instance of the CIP object     | ●                                | ●        | -   | "P.10-42" |
| Set_Attributes_All    | Set all attributes of a specific instance          | This is a function block that sets all attributes of a specific instance of the CIP object          | ●                                | ●        | -   | "P.10-43" |
| NOP                   | NOP service  | This is a function block that executes the NOP service of a specific instance of the CIP object     | ●                                | ●        | -   | "P.10-44" |
| Reset                 | Reset service                                      | This is a function block that executes the Reset service of a specific instance of the CIP object   | ●                                | ●        | -   | "P.10-45" |
| Start                 | Start service                                      | This is a function block that executes the Start service of a specific instance of the CIP object   | ●                                | ●        | -   | "P.10-45" |
| Stop                  | Stop service                                       | This is a function block that executes the Stop   | ●                                | ●        | -   | "P.10-46" |

## 1.3 List of Function Block Instructions

| Name          | Function                               | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|---------------|--|--|----------------------------------|----------|---|-----------|
|               |  |  | (●: Supported, -: Not supported) |          |   |           |
|               |  | service of a specific instance of the CIP object |                                  |          |   |           |
| ENIP.ERROR    | Message service instruction error code | -  | ●                                | ●        | -   | "P.10-47" |
| ENIP.CIPClass | Service class code                     | -  | ●                                | ●        | -   | "P.10-49" |

### ■ SD card operation (File operation)

Files in the SD card inserted in the SD memory card slot can be operated.

| Name              | Function             | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|-------------------|----------------------|--|----------------------------------|----------|---|-----------|
|                   |                      |  | (●: Supported, -: Not supported) |          |   |           |
| FILE.Open         | Open file            | Opens a file or creates a new file.  | ●                                | ●        | -   | "P.10-52" |
| FILE.Close        | Close file           | Closes a file.   | ●                                | ●        | -   | "P.10-53" |
| FILE.Read         | Read file            | Reads data from the file opened by the Open instruction.                     | ●                                | ●        | -   | "P.10-53" |
| FILE.Write        | Write file           | Writes data to the file opened by the Open instruction.                      | ●                                | ●        | -   | "P.10-54" |
| FILE.Flush        | Flush file           | Flushes buffer contents to the file opened by the Open instruction.          | ●                                | ●        | -   | "P.10-55" |
| FILE.Copy         | Copy file            | Copies a file.   | ●                                | ●        | -   | "P.10-56" |
| FILE.Rename       | Rename file          | Changes a file name.   | ●                                | ●        | -   | "P.10-57" |
| FILE.Delete       | Delete file          | Deletes a file.  | ●                                | ●        | -   | "P.10-58" |
| FILE.EOF          | EOF of file          | Determines whether the current offset of a file is EOF (End Of File) or not. | ●                                | ●        | -   | "P.10-58" |
| FILE.GetAttribute | Get file attribute   | Gets file attributes (compressed, hidden, normal, read only).                | ●                                | ●        | -   | "P.10-59" |
| FILE.GetPos       | Get file offset      | Gets the current offset of a file.   | ●                                | ●        | -   | "P.10-60" |
| FILE.GetSize      | Get file size        | Gets the file size.  | ●                                | ●        | -   | "P.10-60" |
| FILE.GetTime      | Get file update time | Get the update time of a file  | ●                                | ●        | -   | "P.10-61" |
| FILE.SetPos       | Set file offset      | Sets the offset of a file.   | ●                                | ●        | -   | "P.10-62" |

## 1.3 List of Function Block Instructions

### ■ SD card operation (Directory operation)

Directories in the SD card inserted in the SD memory card slot can be operated.

| Name           | Function         | Overview  | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|----------------|------------------|---|----------------------------------|----------|---|-----------|
|                |                  |   | (●: Supported, -: Not supported) |          |   |           |
| FILE.DirCreate | Create directory | Creates a directory with a specified name.                    | ●                                | ●        | -   | "P.10-64" |
| FILE.DirOpen   | Open directory   | Opens a directory.  | ●                                | ●        | -   | "P.10-64" |
| FILE.DirClose  | Close directory  | Closes a directory  | ●                                | ●        | -   | "P.10-65" |
| FILE.DirCopy   | Copy directory   | Copies a directory.   | ●                                | ●        | -   | "P.10-66" |
| FILE.DirRename | Rename directory | Renames a directory   | ●                                | ●        | -   | "P.10-67" |
| FILE.DirRemove | Delete directory | Deletes a directory.  | ●                                | ●        | -   | "P.10-67" |
| FILE.DirList   | Directory list   | Outputs a list of directories and files inside the directory. | ●                                | ●        | -   | "P.10-68" |

### ■ Clock setting

The following table lists the function blocks that are used to set the clock of the GM1 Controller.

| Name            | Function                            | Overview   | RTEX                             | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|-----------------|-------------------------------------|--|----------------------------------|----------|---|-----------|
|                 |                                     |  | (●: Supported, -: Not supported) |          |   |           |
| SYS_GetTime     | Get time                            | This is a function block that gets the current local time                | ●                                | ●        | -   | "P.10-70" |
| SYS_SetTime     | Set time                            | This is a function block that sets the current local time.               | ●                                | ●        | -   | "P.10-70" |
| SYS_GetTimezone | Get time zone information           | This is a function block that gets the time zone information.            | ●                                | ●        | -   | "P.10-71" |
| SYS_SetTimezone | Set time zone information           | This is a function block that sets the time zone information.            | ●                                | ●        | -   | "P.10-71" |
| SYS_DateConcat  | Convert from UINT type to DATE type | This is a function that converts a UINT type date to a DATE type.        | ●                                | ●        | -   | "P.10-72" |
| SYS_DateSplit   | Convert from DATE type to UINT type | This is a function that converts a DATE type date to a UINT type.        | ●                                | ●        | -   | "P.10-73" |
| SYS_DTConcat    | Convert from UINT type to DT type   | This is a function that converts a UINT type date and time to a DT type. | ●                                | ●        | -   | "P.10-73" |

## 1.3 List of Function Block Instructions

| Name             | Function                           | Overview   | RET                              | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|------------------|------------------------------------|--|----------------------------------|----------|---|-----------|
|                  |                                    |  | (●: Supported, -: Not supported) |          |   |           |
| SYS_DTSplit      | Convert from UINT type to DT type  | This is a function that converts a UINT type date and time to a DT type.           | ●                                | ●        | -   | "P.10-74" |
| SYS_GetDayOfWeek | Get day of the week                | This is a function that gets the day of the week from the DATE type date.          | ●                                | ●        | -   | "P.10-75" |
| SYS_TODConcat    | Convert from UINT type to TOD type | This is a function that converts a UINT type time with milliseconds to a TOD type. | ●                                | ●        | -   | "P.10-75" |
| SYS_TODSplit     | Convert from UINT type to TOD type | This is a function that converts a TOD type time with milliseconds to a UINT type. | ●                                | ●        | -   | "P.10-76" |
| ERROR            | Clock instruction error code       | -  | ●                                | ●        | -   | "P.10-77" |

### ■ System data

| Name               | Function         | Overview  | RET                              | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|--------------------|------------------|---|----------------------------------|----------|---|-----------|
|                    |                  |   | (●: Supported, -: Not supported) |          |   |           |
| SYS_GetSystemError | Get system error | Gets the information of a system error that has occurred in the GM1 Controller. | ●                                | ●        | -   | "P.10-78" |

### ■ PID control

The following table lists the function blocks that are used to control PID of the GM1 Controller.

| Name         | Function                     | Overview  | RET                              | EtherCAT | Simulation (●: Supported, -: Not supported) | Page      |
|--------------|------------------------------|---|----------------------------------|----------|---|-----------|
|              |                              |   | (●: Supported, -: Not supported) |          |   |           |
| PD           | PD control                   | Performs PD control.                                  | ●                                | ●        | -   | "P.10-79" |
| PID          | PID control                  | Performs PID control.                                 | ●                                | ●        | -   | "P.10-80" |
| PID_FIXCYCLE | PID control (any cycle time) | Performs PID control. Cycle time can be manually set. | ●                                | ●        | -   | "P.10-81" |

### 1.3.8 Function Blocks (For the GM1 Pulse Output Unit)

The following table lists the function blocks used to control the GM1 Pulse Output Unit.

### 1.3 List of Function Block Instructions

| Name              | Function                                    | Overview  | RETX                             | EtherCAT | Simulation | Page      |
|-------------------|---|---|----------------------------------|----------|------------|-----------|
|                   |   |   | (●: Supported, —: Not supported) |          |            |           |
| PG_Power          | Servo ON or OFF                             | Performs servo ON/OFF control.  | ●                                | ●        | -          | "P.11-4"  |
| PG_Jog            | Jogging                                     | Causes the axis to keep traveling in a forward or backward direction.                     | ●                                | ●        | -          | "P.11-5"  |
| PG_MoveAbsolute   | Absolute value positioning                  | Causes the axis to travel to a position specified as an absolute position.                | ●                                | ●        | -          | "P.11-6"  |
| PG_MoveRelative   | Relative value positioning                  | Causes the axis to travel to a position specified as a relative position.                 | ●                                | ●        | -          | "P.11-7"  |
| PG_LatchPosition  | Latch relative positioning                  | Causes the axis to travel to the relative position specified by an external signal input. | ●                                | ●        | -          | "P.11-8"  |
| PG_Pulser         | Pulser operation                            | Enables constant speed operation for the axes using an external pulse input.              | ●                                | ●        | -          | "P.11-10" |
| PG_Stop           | Forced stop and deceleration stop           | Causes the axis to make a forced stop or deceleration stop                                | ●                                | ●        | -          | "P.11-11" |
| PG_Home           | Home return                                 | Causes the axis to make a home return.  | ●                                | ●        | -          | "P.11-12" |
| PG_SetPosition    | Elapsed value and feedback counter settings | Sets the elapsed value and the feedback counter to desired values.                        | ●                                | ●        | -          | "P.11-13" |
| PG_WriteParameter | Write parameters                            | Writes the parameters to the pulse output unit.   | ●                                | ●        | -          | "P.11-14" |
| PG_ReadParameter  | Read parameters                             | Reads the parameters from the pulse output unit.  | ●                                | ●        | -          | "P.11-17" |
| PG_ClearError     | Clear errors                                | Clears the limit error or the set value error of the pulse output unit.                   | ●                                | ●        | -          | "P.11-18" |
| PG_ReadStatus     | Read status                                 | Reads the status from the pulse output unit.  | ●                                | ●        | -          | "P.11-18" |



### 1.4 List of Function Block Instructions that Cannot Be Used with the GM1

#### ■ Instructions not available for Modbus

The following function blocks in the IoDrvModbusTCP, IoDrvModbusTCP Slave, IoDrvModbus, and IoDrvModbusSerialSlave libraries are not available for the GM1 Controller.

| Name                     | Function | Alternative function | Page |
|--------------------------|----------|----------------------|------|
| ModbusTCP SlaveBase      | -        | -                    | -    |
| ModbusTCP SlaveUnit      | -        | -                    | -    |
| ModbusTCP SlaveUnit_Diag | -        | -                    | -    |
| IoDrvModbusTCP_Diag      | -        | -                    | -    |
| ModbusTCP Slave_Diag     | -        | -                    | -    |
| ModbusTCP DeviceDiag     | -        | -                    | -    |
| IoDrvModbusComPort_Diag  | -        | -                    | -    |
| ModbusSlaveComPort_Diag  | -        | -                    | -    |
| IoDrvModbusSerialSlave   | -        | -                    | -    |
| ModbusSerialDeviceDiag   | -        | -                    | -    |
| ModbusServer             | -        | -                    | -    |

#### ■ Instructions not available for general-purpose communication

The following function blocks in the CAA NBS(Net Base Services) library are not available for the GM1 Controller.

| Name              | Function | Alternative function | Page |
|-------------------|----------|----------------------|------|
| TCP_ReadBuffer    | -        | -                    | -    |
| TCP_WriteBuffer   | -        | -                    | -    |
| UDP_ReceiveBuffer | -        | -                    | -    |
| UDP_SendBuffer    | -        | -                    | -    |
| DummyJob          | -        | -                    | -    |

#### ■ Instructions not available for EtherNet/IP

The following function blocks in the IoDrvEtherNetIP and IoDrvEtherNetIPAdapter libraries are not available for the GM1 Controller.

| Name                        | Function | Alternative function | Page |
|-----------------------------|----------|----------------------|------|
| IoDrvEtherNetIP_diag        | -        | -                    | -    |
| RemoteAdapter_diag          | -        | -                    | -    |
| AdapterDiagnosis            | -        | -                    | -    |
| IoDrvEtherNetIPAdapter_Diag | -        | -                    | -    |
| Module_Diag                 | -        | -                    | -    |

## 1.4 List of Function Block Instructions that Cannot Be Used with the GM1

### ■ Instructions not available for motion control

The following function blocks in the SM3\_Basic library are not available for the GM1 Controller. Alternative functions are listed, if available.

| Name                           | Function   | Alternative function                        | Page    |
|--------------------------------|--|---|---------|
| SMC_Commissioning              | Commissioning status                             | Commissioning function of the GM Programmer | -       |
| SMC_SetCustomRampType          | Set acceleration / deceleration custom operation | -   | -       |
| SMC_CAM_ObjectManager          | Manage cam data                                  | -   | -       |
| SMC3_CommunicateDriveParameter | Communication setting                            | RTEX_ReadAmpParameter                       | "P.8-7" |
| SMC3_ReadDriveParameter        | Read drive parameter                             | RTEX_ReadAmpParameter                       | "P.8-7" |
| SMC3_ReadParameter             | Read parameter                                   | RTEX_ReadAmpParameter                       | "P.8-7" |
| SMC3_WriteDriveParameter       | Write drive parameter                            | RTEX_WriteAmpParameter                      | "P.8-8" |
| SMC3_WriteParameter            | Write parameter                                  | RTEX_WriteAmpParameter                      | "P.8-8" |
| MC_ReadBoolParameter           | Read BOOL-type parameter                         | RTEX_ReadAmpParameter                       | "P.8-7" |
| MC_ReadParameter               | Read parameter                                   | RTEX_ReadAmpParameter                       | "P.8-7" |
| MC_WriteBoolParameter          | Write BOOL-type parameter                        | RTEX_WriteAmpParameter                      | "P.8-8" |
| MC_WriteParameter              | Write parameter                                  | RTEX_WriteAmpParameter                      | "P.8-8" |
| SMC_VIRTUAL_AXIS               | Set virtual axis                                 | -   | -       |
| SMC3_BrakeStatus               | Get brake status                                 | -   | -       |
| SMC3_BrakeControl              | Brake control                                    | -   | -       |
| SMC3_PersistPosition           | Persist actual axis position                     | -   | -       |
| SMC3_PersistPositionLogical    | Persist logical axis position                    | -   | -       |
| SMC3_PersistPositionSingleturn | Persist actual axis position with a range        | -   | -       |
| SMC_PerfStat                   | Calculate performance statistics                 | -   | -       |
| SMC_SeriesStat                 | Calculate increment statistics                   | -   | -       |
| SMC_AxisDiagnosticLog          | Log axis parameter                               | -   | -       |
| FB_Template_Edge               |  | -   | -       |
| FB_Template_EdgeAbort          |  | -   | -       |
| FB_Template_EdgeAbortTimeout   |  | -   | -       |
| SMC_StartupDrive               |  | -   | -       |
| SMC_ChangeDynamicLimits        |  | -   | -       |
| SMC_ChangeGearingRatio         |  | -   | -       |
| SMC_SetMovementType            |  | -   | -       |
| SMC_SetRampType                |  | -   | -       |
| SMC_SetSoftwareLimits          |  | -   | -       |
| SMC_CAMBounds                  |  | -   | -       |

## 1.4 List of Function Block Instructions that Cannot Be Used with the GM1

---

| Name                               | Function                  | Alternative function | Page |
|------------------------------------|---------------------------|----------------------|------|
| SMC_CAMBounds_Pos                  |                           | -                    | -    |
| SMC_CamEditor                      |                           | -                    | -    |
| SMC_CamRegister                    |                           | -                    | -    |
| SMC_GetCamSlaveSetPosition         |                           | -                    | -    |
| SMC_ReadCAM                        |                           | -                    | -    |
| SMC_WriteCAM                       |                           | -                    | -    |
| SMC_PerfTimerSum                   |                           | -                    | -    |
| SMC_FollowPosition                 |                           | -                    | -    |
| SMC_FollowPositionVelocity         |                           | -                    | -    |
| SMC_FollowSetValues                |                           | -                    | -    |
| SMC_FollowVelocity                 |                           | -                    | -    |
| SMC_Homing                         |                           | -                    | -    |
| ETC_CO_SdoInfoGeEntryDescription   | Read object name          | -                    | -    |
| ETC_CO_SdoInfoGetODList            | Read object tree          | -                    | -    |
| ETC_CO_SdoInfoGetObjectDescription | Read object information   | -                    | -    |
| ReadEEPromData                     | Read slave EEPROM         | -                    | -    |
| ReadWriteEEProm                    | Read / write slave EEPROM | -                    | -    |

(MEMO)

# 2 Ladder Instructions

---

|  |     |
|--|-----|
| 2.1 Ladder Instructions .....              | 2-2 |
| 2.1.1 NO Contact .....                     | 2-2 |
| 2.1.2 NC Contact .....                     | 2-3 |
| 2.1.3 Rising Edge Detection Contact .....  | 2-3 |
| 2.1.4 Falling Edge Detection Contact ..... | 2-4 |
| 2.1.5 Parallel NO Contact .....            | 2-5 |
| 2.1.6 Parallel NC Contact .....            | 2-6 |
| 2.1.7 Coil .....                           | 2-7 |
| 2.1.8 Negated Coil .....                   | 2-7 |
| 2.1.9 Set Coil .....                       | 2-8 |
| 2.1.10 Reset Coil .....                    | 2-9 |

## 2.1 Ladder Instructions

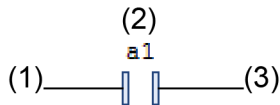
### 2.1 Ladder Instructions

This section describes ladder instructions that can be used for ladder diagram program (LD program).

#### 2.1.1 NO Contact

If the variable corresponding to the contact is TRUE, then the input value is output. If the variable is FALSE, then FALSE is output.

##### ■ Icon




##### ■ Parameter

| No. | Scope         | Type | Description                                 |
|-----|---------------|------|---|
| (1) | Input         | BOOL | Input to the NO contact                     |
| (2) | Variable name | BOOL | Variable that corresponds to the NO contact |
| (3) | Output        | BOOL | Output from the NO contact                  |

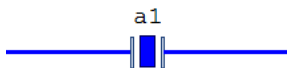
##### ■ Input method

Use one of the following methods to input the NO contact.

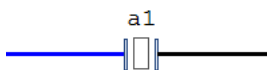
- From the tool box, select **Ladder elements>NO contact** and drag to "Start from here".
- Right-click on the network, and, from the displayed menu, select **Insert Contact**.
- Click the  icon on the toolbar.
- From the menu, select **FBD / LD / IL>Insert Contact**.
- Press the shortcut keys <Ctrl+k> simultaneously.

##### ■ Program example

If the variable (a1) corresponding to the NO contact is TRUE, then the value input to the NO contact (TRUE) is output as is.



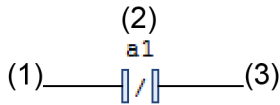
If the variable (a1) corresponding to the contact is FALSE, then FALSE is output.



**2.1.2 NC Contact**

If the variable corresponding to the contact is TRUE, then FALSE is output. If the variable is FALSE, then the input value is output.

■ **Icon**




■ **Parameter**

| No. | Scope         | Type | Description                                 |
|-----|---------------|------|---|
| (1) | Input         | BOOL | Input to the NC contact                     |
| (2) | Variable name | BOOL | Variable that corresponds to the NC contact |
| (3) | Output        | BOOL | Output from the NC contact                  |

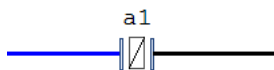
■ **Input method**

Use one of the following methods to input the NC contact.

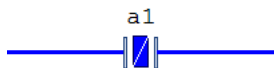
- From the tool box, select **Ladder elements>NC contact** and drag to "Start from here".
- Right-click on the network, and, from the displayed menu, select "Insert NC contact".
- Click the  icon on the toolbar.
- From the menu, select **FBD / LD / IL>Insert NC contact**.

■ **Program example**

If the variable (a1) corresponding to the NC contact is TRUE, then FALSE is output.



If the variable (a1) corresponding to the NC contact is FALSE, then the value input to the NC contact (TRUE) is output as is.

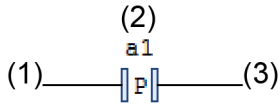


**2.1.3 Rising Edge Detection Contact**

If a rising edge is detected in the variable corresponding to the contact, then the input value is output for one cycle only.

## 2.1 Ladder Instructions

### ■ Icon




### ■ Parameter

| No. | Scope         | Type | Description  |
|-----|---------------|------|--|
| (1) | Input         | BOOL | Input to the contact   |
| (2) | Variable name | BOOL | Variable that corresponds to the rising edge detection contact |
| (3) | Output        | BOOL | Output from the contact  |

### ■ Input method

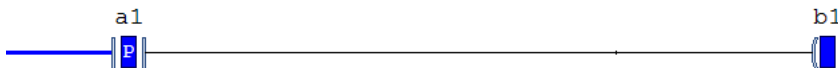
Input the rising edge detection contact by first inputting the NO contact and then changing the NO contact.

Select the NO contact just input and then perform one of the following operations.

- Right-click and, from the displayed menu, select **Edge detection** .
- From the menu, select **FBD / LD / IL>Edge detection** .
- Press the shortcut keys <Ctrl+e> simultaneously.
- Click the  icon on the toolbar.

### ■ Program example

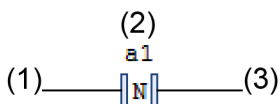
The following program is designed to detect the rising edge with the variable (a1) corresponding to the rising edge detection contact and to output TRUE for one cycle only.



### 2.1.4 Falling Edge Detection Contact

If a falling edge is detected in the variable corresponding to the contact, then the input value is output for one cycle only.

### ■ Icon



### ■ Parameter

| No. | Scope | Type | Description          |
|-----|-------|------|----------------------|
| (1) | Input | BOOL | Input to the contact |




| No. | Scope         | Type | Description   |
|-----|---------------|------|---|
| (2) | Variable name | BOOL | Variable that corresponds to the falling edge detection contact |
| (3) | Output        | BOOL | Output from the contact   |

### ■ Input method

Input the falling edge detection contact by first inputting the NO contact and then changing the NO contact.

Select the NO contact just input and then perform one of the following operations.

- Right-click and, from the displayed menu, select **Edge detection** twice.
- From the menu, select **FBD / LD / IL>Edge detection** twice.
- Press the shortcut keys Ctrl+e simultaneously twice.
- Click the  icon on the toolbar twice.

### ■ Program example

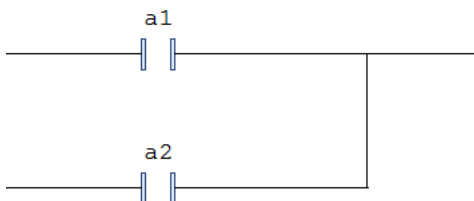
The following program is designed to detect the falling edge with the variable (a1) corresponding to the falling edge detection contact and to output TRUE for one cycle only.



## 2.1.5 Parallel NO Contact

NO contacts can be input in parallel to the initial contact. Of the contacts wired in parallel, if the output of one or more contacts is TRUE, TRUE is output.


### ■ Icon



### ■ Input method

To input a parallel NO contact below the initial contact, select **Ladder elements >Parallel NO contact** from the tool box and drag to the position indicated with “▼” next to the contact.

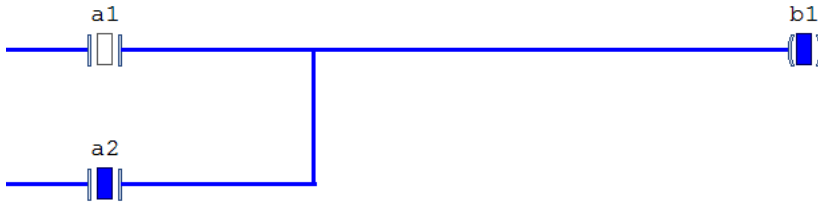
Or, with the contact selected, perform one of the following operations.

- Right-click, and, from the displayed menu, select **Insert contact in parallel (below)**.
- From the menu, select **FBD/LD/IL>Insert contact in parallel (below)**.
- Press the shortcut keys <Ctrl+r> simultaneously.
- Click the  icon on the toolbar.

## 2.1 Ladder Instructions

### ■ Program example

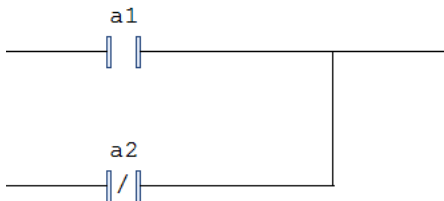
This program is designed to input one NO contact in parallel to the NO contact. TRUE is output because the NO contact below is TRUE.



### 2.1.6 Parallel NC Contact

NC contacts can be input in parallel to the initial contact. Of the contacts wired in parallel, if the output of one or more contacts is TRUE, TRUE is output.


### ■ Icon



### ■ Input method

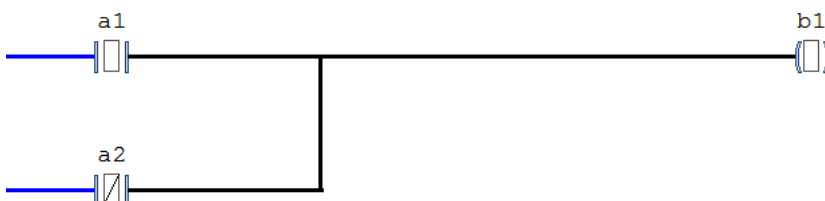
To input a parallel NC contact below the initial contact, select **Ladder elements >Parallel NC contact** from the tool box and drag to the position indicated with “▼” next to the contact.

Or, with the contact selected, perform one of the following operations.

- Right-click, and, from the displayed menu, select "Insert NC contact in parallel (below)".
- From the menu, select **FBD / LD / IL>Insert NC contact in parallel (below)**.
- Click the  icon on the toolbar.

### ■ Program example

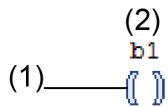
This program is designed to input one NC contact in parallel to the NO contact. FALSE is output because the outputs of both contacts are FALSE.



### 2.1.7 Coil

The input value is saved in the variable corresponding to the coil. If the input value is TRUE, then TRUE is saved. If the input value is FALSE, then FALSE is saved.

#### ■ Icon




#### ■ Parameter

| No. | Scope         | Type | Description                                       |
|-----|---------------|------|---|
| (1) | Input         | BOOL | Input to the coil                                 |
| (2) | Variable name | BOOL | Name of the variable that corresponds to the coil |

#### ■ Input method

Use one of the following methods to input the coil.

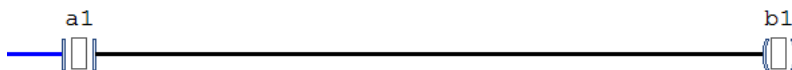
- From the tool box, select **Ladder elements> Coil** and drag to "Add output or jump here" (when connecting to a contact).
- Right-click on the network, and, from the displayed menu, select **Insert Coil**.
- Click the icon on the tool bar.
- From the menu, select **FBD / LD / IL>Insert Coil**.
- Press the shortcut keys <Ctrl+a> simultaneously.
- Click the  icon on the toolbar.

#### ■ Program example

This program is designed to input the output from the NO contact to the coil. TRUE is saved in the variable (b1) because the input to the coil is TRUE.



FALSE is saved in the variable (b1) because the input to the coil is FALSE.

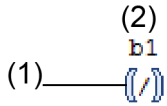


### 2.1.8 Negated Coil

The negated value of the input is saved in the variable corresponding to the coil. If the input value is TRUE, then FALSE is saved. If the input value is FALSE, then TRUE is saved.

## 2.1 Ladder Instructions

### ■ Icon




### ■ Parameter

| No. | Scope         | Type | Description   |
|-----|---------------|------|---|
| (1) | Input         | BOOL | Input to the negated coil                                 |
| (2) | Variable name | BOOL | Name of the variable that corresponds to the negated coil |

### ■ Input method

The negated coil can be input by inputting a coil and changing it. With the input coil selected, perform one of the following operations.

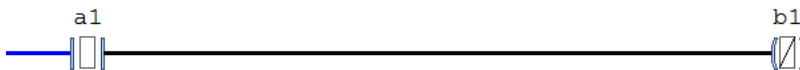
- Right-click and, from the displayed menu, select **Negation**.
- From the menu, select **FBD / LD / IL>Negation**.
- Press the shortcut keys <Ctrl+n> simultaneously.
- Click the  icon on the toolbar.

### ■ Program example

This program is designed to input the output from the NO contact to the negated coil. FALSE is saved in the variable (b1) because the input to the coil is TRUE.



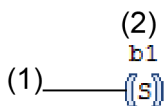
TRUE is saved in the variable (b1) because the input to the coil is FALSE.



### 2.1.9 Set Coil

When the input value turns TRUE, TRUE is saved in the variable corresponding to the coil. TRUE is held until the input to the reset coil that corresponds to the same variable turns TRUE.

### ■ Icon




■ **Parameter**

| No. | Scope         | Type | Description   |
|-----|---------------|------|---|
| (1) | Input         | BOOL | Input to the set coil.                                |
| (2) | Variable name | BOOL | Name of the variable that corresponds to the set coil |

■ **Input method**

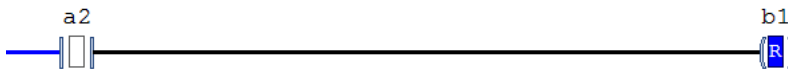
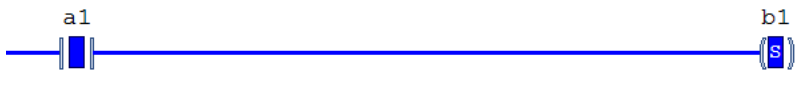
Use one of the following methods to input the set coil.

- From the tool box, select **Ladder elements>Set Coil** and drag to "Add output or jump here" (when connecting to a contact).
- Right-click on the network, and, from the displayed menu, select "Insert Set Coil".
- Click the  icon on the toolbar.
- From the menu, select **FBD / LD / IL>Insert Set Coil**.

■ **Program example**

This program is designed to input the output from the NO contact to the set coil and the reset coil.

TRUE is saved in the set coil variable (b1) because the input to the set coil is TRUE.



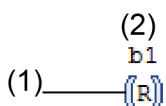
 **Info.**

- Each set coil should be accompanied by a reset coil.

**2.1.10 Reset Coil**

When the input value turns TRUE, FALSE is saved in the variable corresponding to the coil. FALSE is held until the input to set coil that corresponds to the same variable turns TRUE.

■ **Icon**



## 2.1 Ladder Instructions


---

### ■ Parameter

| No. | Scope         | Type | Description   |
|-----|---------------|------|---|
| (1) | Input         | BOOL | Input to the reset coil.                                |
| (2) | Variable name | BOOL | Name of the variable that corresponds to the reset coil |

### ■ Input method

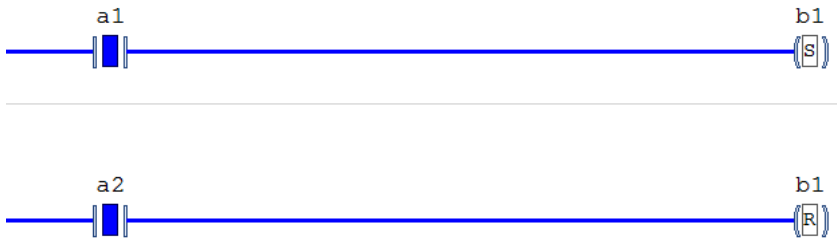
Use one of the following methods to input the reset coil.

- From the tool box, select **Ladder elements>Reset Coil** and drag to "Add output or jump here" (when connecting to a contact).
- Right-click on the network, and, from the displayed menu, select **Reset Coil**.
- Click the  icon on the toolbar.
- From the menu, select **FBD / LD / IL>Insert Reset Coil**.

### ■ Program example

This program is designed to input the output from the NO contact to the set coil and the reset coil.

FALSE is saved in the variable (b1) because the input to the reset coil is TRUE.



### Info.

- Each set coil should be accompanied by a reset coil.

# 3 Functions

---

|   |      |
|---|------|
| 3.1 Substitution Instruction.....                   | 3-3  |
| 3.1.1 MOVE (Substitution) .....                     | 3-3  |
| 3.2 Arithmetic Operation Instructions.....          | 3-4  |
| 3.2.1 ADD (Addition) .....                          | 3-4  |
| 3.2.2 SUB (Subtraction) .....                       | 3-5  |
| 3.2.3 MUL (Multiplication) .....                    | 3-6  |
| 3.2.4 DIV (Division) .....                          | 3-7  |
| 3.2.5 MOD (Remainder).....                          | 3-8  |
| 3.3 Boolean Operation Instructions.....             | 3-9  |
| 3.3.1 AND (Logical AND) .....                       | 3-9  |
| 3.3.2 OR (Logical OR) .....                         | 3-10 |
| 3.3.3 NOT (Negation).....                           | 3-10 |
| 3.3.4 XOR (Exclusive OR) .....                      | 3-11 |
| 3.3.5 AND_THEN (Logical AND) .....                  | 3-12 |
| 3.3.6 OR_ELSE (Logical OR) .....                    | 3-13 |
| 3.4 Comparison Operation Instructions .....         | 3-15 |
| 3.4.1 EQ ("Equal" Comparison) .....                 | 3-15 |
| 3.4.2 NE ("Not Equal" Comparison) .....             | 3-15 |
| 3.4.3 LT ("Less Than" Comparison) .....             | 3-16 |
| 3.4.4 LE ("Less Than or Equal" Comparison) .....    | 3-17 |
| 3.4.5 GT ("Greater Than" Comparison) .....          | 3-18 |
| 3.4.6 GE ("Greater Than Or Equal" Comparison) ..... | 3-19 |
| 3.5 Bit Shift Instructions .....                    | 3-21 |
| 3.5.1 SHL (Shift Left) .....                        | 3-21 |
| 3.5.2 SHR (Shift Right) .....                       | 3-22 |
| 3.5.3 ROL (Rotate Left).....                        | 3-22 |
| 3.5.4 ROR (Rotate Right).....                       | 3-23 |
| 3.6 Selection Instructions.....                     | 3-25 |
| 3.6.1 SEL (Binary Selector) .....                   | 3-25 |
| 3.6.2 MUX (Multiplexer) .....                       | 3-26 |
| 3.6.3 LIMIT (Limiter).....                          | 3-27 |
| 3.6.4 MAX (Maximum Value) .....                     | 3-28 |
| 3.6.5 MIN (Minimum Value).....                      | 3-28 |
| 3.7 Numerical Operation Instructions.....           | 3-30 |
| 3.7.1 ABS (Absolute Value).....                     | 3-30 |
| 3.7.2 SQRT (Square Root).....                       | 3-30 |
| 3.7.3 LN (Natural Logarithm) .....                  | 3-31 |
| 3.7.4 LOG (Common Logarithm) .....                  | 3-32 |

## 3 Functions

---

|  |      |
|--|------|
| 3.7.5 EXP (Natural Exponent).....                                  | 3-33 |
| 3.7.6 EXPT (Exponentiation) .....                                  | 3-33 |
| 3.7.7 SIN (Trigonometric Function Sine).....                       | 3-34 |
| 3.7.8 COS (Trigonometric Function Cosine) .....                    | 3-35 |
| 3.7.9 TAN (Trigonometric Function Tangent) .....                   | 3-36 |
| 3.7.10 ASIN (Trigonometric Function Arc Sine).....                 | 3-37 |
| 3.7.11 ACOS (Trigonometric Function Arc Cosine) .....              | 3-38 |
| 3.7.12 ATAN (Trigonometric Function Arc Tangent).....              | 3-38 |
| 3.8 Data Type Conversion Instructions .....                        | 3-40 |
| 3.8.1 "Type 1" _TO_ "Type 2" ("Type 1">"Type 2" Conversion).....   | 3-40 |
| 3.8.2 TRUNC (Real Number to DINT Conversion) .....                 | 3-41 |
| 3.8.3 TRUNC_INT (Real Number to INT Conversion) .....              | 3-41 |
| 3.9 Character string instructions .....                            | 3-43 |
| 3.9.1 LEN (Length of a Character String) .....                     | 3-43 |
| 3.9.2 LEFT (Extracting Characters from the Left End).....          | 3-43 |
| 3.9.3 RIGHT (Extracting Characters from the Right End).....        | 3-44 |
| 3.9.4 MID (Extracting Characters from the Specified Position)..... | 3-45 |
| 3.9.5 CONCAT (Concatenating Character Strings).....                | 3-46 |
| 3.9.6 INSERT (Inserting a Character String).....                   | 3-47 |
| 3.9.7 DELETE (Deleting a Character String) .....                   | 3-48 |
| 3.9.8 REPLACE (Replacing a Character String).....                  | 3-49 |
| 3.9.9 FIND (Search for Character String) .....                     | 3-50 |
| 3.10 Other Instructions.....                                       | 3-52 |
| 3.10.1 SIZEOF (Get the Size).....                                  | 3-52 |
| 3.10.2 ADR (Get the Address) .....                                 | 3-52 |
| 3.11 SD Memory Card Slot Instruction.....                          | 3-54 |
| 3.11.1 SYS_GetSDCoverState (Get SD Card Cover Open / Close State)  | 3-54 |
| 3.11.2 SYS_GetSDAccessRdy (Get SD Card Access Ready State).....    | 3-54 |



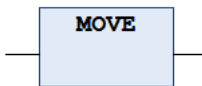
## 3.1 Substitution Instruction

Use the substitution instruction to substitute a variable with a value of another variable.

### 3.1.1 MOVE (Substitution)

This is a function that substitutes the value of a variable specified in the input for a variable specified in the output.

#### ■ Icon



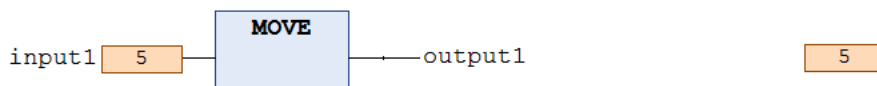
#### ■ Parameter

| Scope  | Type | Description  |
|--------|------|--|
| Input  | All  | Specifies the variable of the substitution source. |
| Output | All  | Specifies the variable of the substitution target. |

#### ■ Program example

This program is designed to substitute the value of input variable “input1” for the output variable “output1”.

#### LD program



#### ST program

```
output1 [ 5 ] := MOVE (input1 [ 5 ] );
```

It is also possible to substitute the value using an operator (:=).

```
output1 [ 5 ] := input1 [ 5 ] ;
```

## 3.2 Arithmetic Operation Instructions

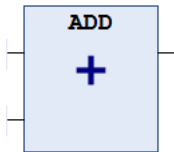
### 3.2 Arithmetic Operation Instructions

Arithmetic operation instructions can be used to perform calculation such as four arithmetic operations.

#### 3.2.1 ADD (Addition)

This is a function that adds input arguments and outputs the sum.

##### ■ Icon



##### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the variables to be added.                 |
| Output | (Note 1) | Outputs the sum of variables specified in the input. |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL, TIME, TIME\_OF\_DAY, DATE\_AND\_TIME

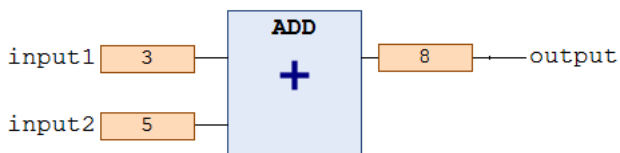
Time type data can be added in the following combinations.

- TIME + TIME = TIME
- TIME\_OF\_DAY + TIME = TIME\_OF\_DAY
- DATE\_AND\_TIME + TIME = DATE\_AND\_TIME

##### ■ Program example

This program is designed to output the sum of input variables “input1” and “input2” to the output variable “output”.

##### LD program



##### ST program

It is possible to add the values using “+” operator.

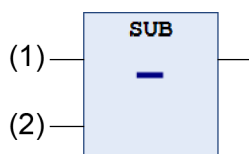
```
output[ 8 ] := input1[ 3 ] + input2[ 5 ];
```

**i Info.**

- If you want to increase input arguments in the LD program, right-click on the ADD function, and, on the displayed menu, select "Add Input".

**3.2.2 SUB (Subtraction)**

This is a function that subtracts input arguments and outputs the difference.

**■ Icon****■ Parameter**

| Scope  | Number   | Type     | Description   |
|--------|----------|----------|---|
| Input  | (1), (2) | (Note 1) | Specifies the variables to be subtracted.                                   |
| Output | -        | (Note 1) | Outputs the value obtained by subtracting the input (2) from the input (1). |

(Note 1) Usable data types

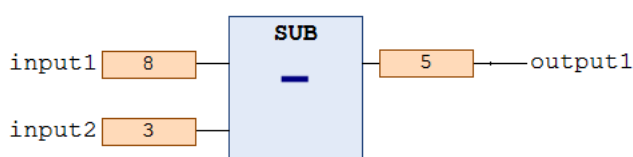
BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL, TIME, TIME\_OF\_DAY, DATE, DATE\_AND\_TIME

For time type data, subtraction can be performed in the following combinations. Note that negative time cannot be calculated.

- TIME - TIME = TIME
- DATE - DATE = TIME
- TOD - TIME = TOD
- TOD - TOD = TIME
- DT - TIME = DT
- DT - DT = TIME

**■ Program example**

This program is designed to output the difference between the input variables "input1" and "input2" to the output variable "output1".

**LD program**

## 3.2 Arithmetic Operation Instructions

### ST program

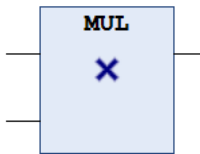
It is possible to subtract the values using “-” operator.

```
output1 [ 5 ] := input1 [ 8 ] - input2 [ 3 ] ;
```

### 3.2.3 MUL (Multiplication)

This is a function that multiplies input arguments and outputs the product.

#### ■ Icon



#### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the variables to be multiplied.                |
| Output | (Note 1) | Outputs the product of variables specified in the input. |

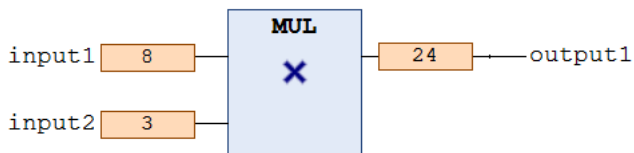
(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL, TIME

#### ■ Program example

This program is designed to output the product of the input variables “input1” and “input2” to the output variable “output1”.

#### LD program



### ST program

It is possible to multiply the values using “\*” operator.

```
output1 [ 24 ] := input1 [ 8 ] * input2 [ 3 ] ;
```

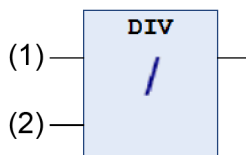
### **i** Info.

- If you want to increase input arguments in the LD program, right-click on the MUL function, and, on the displayed menu, select "Add Input".
- TIME type data cannot be multiplied by REAL type, LREAL type, or TIME type data.

### 3.2.4 DIV (Division)

This is a function that divides input arguments and outputs the quotient.

#### ■ Icon



#### ■ Parameter

| Scope  | No.      | Type     | Description   |
|--------|----------|----------|---|
| Input  | (1), (2) | (Note 1) | Specifies the variables to be divided.                                    |
| Output | -        | (Note 1) | Outputs the quotient obtained by dividing the input (2) by the input (1). |

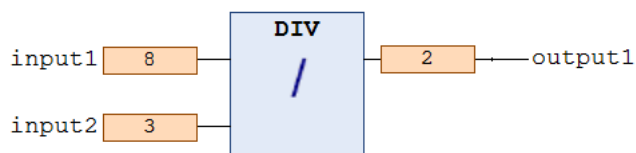
(Note 1) Usable data types

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL, TIME

#### ■ Program example

This program is designed to output the quotient of the INT type input variables "input1" and "input2" to the INT type output variable "output1".

#### LD program



#### ST program

It is possible to divide the values using the division operator ("/").

```
output1[ 2 ] := input1[ 8 ] / input2[ 3 ];
```

## 3.2 Arithmetic Operation Instructions

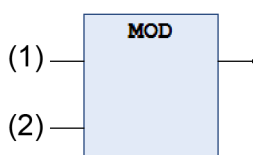
### **i** Info.

- TIME type variables can be divided by integer type variables.
- When a variable is divided by a DINT, LINT, REAL, or LREAL type variable, it can be checked if 0 is used in the calculation. (Refer to “Auto Check POU” in the “SMC Tool Introduction Guide”.)

### 3.2.5 MOD (Remainder)

This is a function that divides input arguments and outputs the remainder.

#### ■ Icon



#### ■ Parameter

| Scope  | No.      | Type     | Description  |
|--------|----------|----------|--|
| Input  | (1), (2) | (Note 1) | Specifies the variables to be divided.                                     |
| Output | -        | (Note 1) | Outputs the remainder obtained by dividing the input (2) by the input (1). |

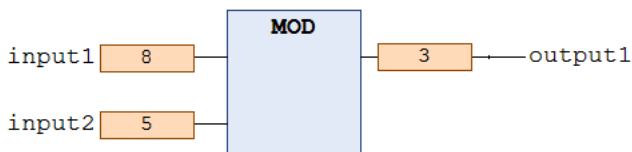
(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT

#### ■ Program example

This program is designed to output the remainder obtained from dividing the INT type input variables “input1” and “input2” to the INT type output variable “output1”.

#### LD program



#### ST program

```
output1 [ 3 ] := input1 [ 8 ] MOD input2 [ 5 ] ;
```

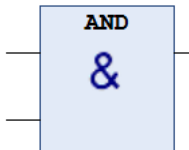
### 3.3 Boolean Operation Instructions

Boolean operation instructions can be used to perform bool operations such as logical AND or logical OR.

#### 3.3.1 AND (Logical AND)

This is a function that outputs logical AND of the input arguments.

##### ■ Icon



##### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the variables to be used to obtain logical AND.        |
| Output | (Note 1) | Outputs the logical AND of the variables specified in the input. |

(Note 1) Usable data type

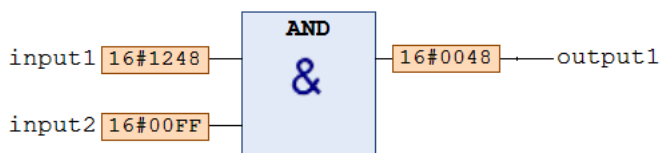
BOOL, BYTE, WORD, DWORD, LWORD

##### ■ Program example

This program is designed to output the logical AND of the WORD type input variables "input1" and "input2" to the output variable "output1".

The execution result is displayed in a hexadecimal number.

##### LD program



##### ST program

```
output1 16#0048 := input1 16#1248 AND input2 16#00FF;
```

##### **i** Info.

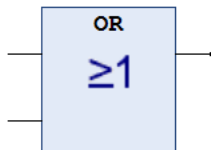
- If you want to increase input arguments in the LD program, right-click on the AND function, and, on the displayed menu, select "Add Input".

## 3.3 Boolean Operation Instructions

### 3.3.2 OR (Logical OR)

This is a function that outputs logical OR of the input arguments.

#### ■ Icon



#### ■ Parameter

| Scope  | Type     | Description   |
|--------|----------|---|
| Input  | (Note 1) | Specifies the variables to be used to obtain logical OR.        |
| Output | (Note 1) | Outputs the logical OR of the variables specified in the input. |

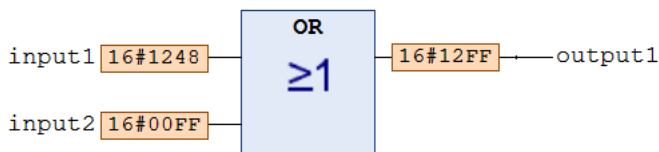
(Note 1) Usable data type  
BOOL, BYTE, WORD, DWORD, LWORD

#### ■ Program example

This program is designed to output the logical OR of the WORD type input variables “input1” and “input2” to the output variable “output1”.

The execution result is displayed in a hexadecimal number.

#### LD program



#### ST program

```
output1 16#12FF := input1 16#1248 OR input2 16#00FF;
```

#### **i** Info.

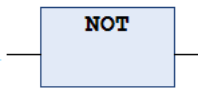
- If you want to increase input arguments in the LD program, right-click on the OR function, and, on the displayed menu, select "Add Input".

### 3.3.3 NOT (Negation)

This is a function that outputs the negation of the input argument.



### ■ Icon



### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the variable to be used to obtain the negation.    |
| Output | (Note 1) | Outputs the negation of the variable specified in the input. |

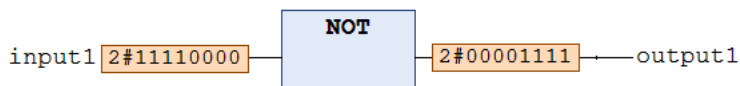
(Note 1) Usable data type  
 BOOL, BYTE, WORD, DWORD, LWORD

### ■ Program example

This program is designed to output the negation of the BYTE type input variable “input1” to the output variable “output1”.

The execution result is displayed in a binary number.

#### LD program



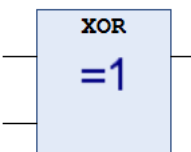
#### ST program

```
output1 2#00001111 := NOT input1 2#11110000 ;
```

### 3.3.4 XOR (Exclusive OR)

This is a function that outputs exclusive OR of the input arguments.

### ■ Icon



### ■ Parameter

| Scope  | Type     | Description   |
|--------|----------|---|
| Input  | (Note 1) | Specifies the variables to be used to obtain exclusive OR.        |
| Output | (Note 1) | Outputs the exclusive OR of the variables specified in the input. |

### 3.3 Boolean Operation Instructions

| Scope | Type | Description   |
|-------|------|---|
|       |      | Outputs 0 if both input bits are 1 or 0.<br>Outputs 1 if one of the two input bits is 1 and the other bit is 0. |

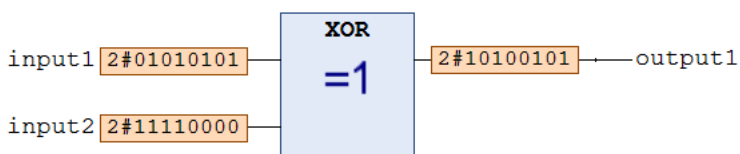
(Note 1) Usable data type  
 BOOL, BYTE, WORD, DWORD, LWORD

#### ■ Program example

This program is designed to output the exclusive OR of the BYTE type input variables “input1” and “input2” to the output variable “output1”.

The execution result is displayed in a binary number.

#### LD program



#### ST program

```
output1 2#10100101 := input1 2#01010101 XOR input2 2#11110000 ;
```

### 3.3.5 AND\_THEN (Logical AND)

This is a conditional AND evaluation function of the input operand.

#### ■ Usable data types

BOOL, BIT

#### ■ Program example

This program is designed to compare the value of the variable accessed by pwAddress (pointer) with wExpected if the pwAddress is not NULL and, if they are the same, substitute with the value of wNewValue.

As default values, “5” is stored in the variable “test1” accessed by pwAddress, “5” in wExpected, and “3” in wNewValue.

As an initial step, judgment is made whether pwAddress is NULL or not. Since it is not NULL, comparison is made between the value of “test1” and the value of wExpected as the next step. Since these two values are both “5”, TRUE is assigned. As a result, the value of wNewValue “3” is stored in the “test1” and the xFlag flag is set to TRUE.

#### ST program

[Declaration section]

```

VAR
  pwAddress   : POINTER TO WORD;
  wExpected   : WORD := 5;
  wNewValue   : WORD := 3;
  xFlag       : BOOL;
  test1       : WORD := 5;
END_VAR

```

[Implementation section]

```

pwAddress[16#F1D10BBE] := ADR(test1[16#0003]);

IF pwAddress[16#F1D10BBE] <> 0 AND THEN pwAddress[16#0003] = wExpected[16#0005] THEN
  pwAddress[16#0003] := wNewValue[16#0003];
  xFlag[TRUE] := TRUE;
ELSE
  xFlag[TRUE] := FALSE;
END_IF

```

### Info.

- Expressions of other operands are executed only when the first operand is TRUE. Therefore, if no value is stored in pwAddress in the above example, the initial NULL judgment turns FALSE. As a result, no judgment is performed on operands after the AND\_THEN operator.

### 3.3.6 OR\_ELSE (Logical OR)

This is a conditional OR evaluation function of the input operand.

#### ■ Usable data types

BOOL, BIT

#### ■ Program example

16#000000FF is stored in the variable dw.

"dw.8" that represents bit 8 of dw is FALSE and "dw.1" that represents bit 1 is TRUE.

Therefore, the operation result flag bX is TRUE.

Note that the third input expression is not executed and bEver remains FALSE.

#### ST program

[Declaration section]

```

VAR
  bEver       : BOOL;
  bX          : BOOL;
  dw          : DWORD := 16#000000FF;
END_VAR

```

[Implementation section]

```

bEver[FALSE] := FALSE;
bX[TRUE] := dw[16#000000FF].0[FALSE] OR_ELSE dw[16#000000FF].1[TRUE] OR_ELSE dw[16#000000FF].2[TRUE] OR_ELSE (bEver[FALSE] := TRUE);

```

### 3.3 Boolean Operation Instructions

---

#### Info.

- In case of OR\_ELSE, when one of the operands is evaluated TRUE, all other operator expressions are not evaluated.

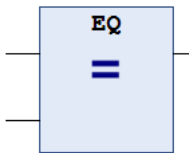
### 3.4 Comparison Operation Instructions

Comparison operation instructions can be used to compare two arguments.

#### 3.4.1 EQ (“Equal” Comparison)

This is a function that compares two input arguments and determines if they are the same value.

##### ■ Icon



##### ■ Parameter

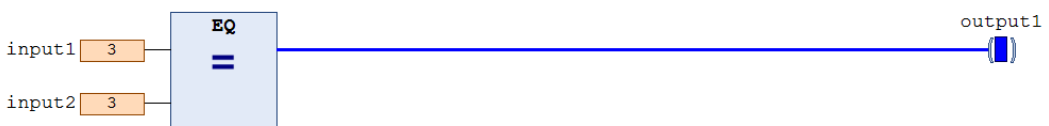
| Scope  | Type | Description   |
|--------|------|---|
| Input  | All  | Specifies the variables to be compared.   |
| Output | BOOL | Outputs TRUE if the input variable values are the same.<br>Outputs FALSE if they are different. |

##### ■ Program example

This program is designed to compare the input variables “input1” and “input2” and output the result to the output variable “output1”.

##### LD program

TRUE is output because the input variable values “input1” and “input2” are the same.



##### ST program

Use the operator (=) to compare the values.

FALSE is output because the input variable values “input1” and “input2” are different.

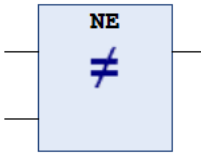
```
output1 FALSE := (input1 3 = input2 5);
```

#### 3.4.2 NE (“Not Equal” Comparison)

This is a function that compares two input arguments and determines if they are not the same.

## 3.4 Comparison Operation Instructions

### ■ Icon



### ■ Parameter

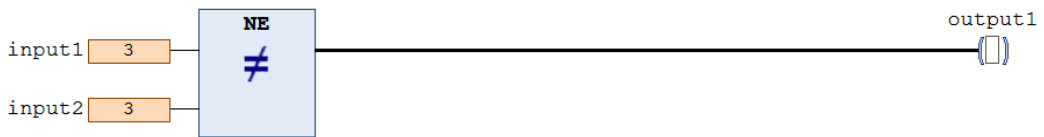
| Scope  | Type | Description   |
|--------|------|---|
| Input  | All  | Specifies the variables to be compared.   |
| Output | BOOL | Outputs TRUE if the input variable values are different.<br>Outputs FALSE if they are the same. |

### ■ Program example

This program is designed to compare the input variables “input1” and “input2” and output the result to the output variable “output1”.

#### LD program

FALSE is output because the input variable values “input1” and “input2” are the same.



#### ST program

Use the operator (<>) to compare the values.

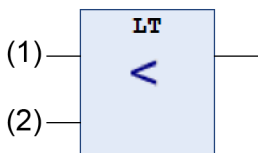
TRUE is output because the input variable values “input1” and “input2” are different.

```
output1 TRUE := input1 3 <> input2 5;
```

### 3.4.3 LT (“Less Than” Comparison)

This is a function that compares two input arguments and determines if the first argument is less than the second argument.

### ■ Icon



### ■ Parameter

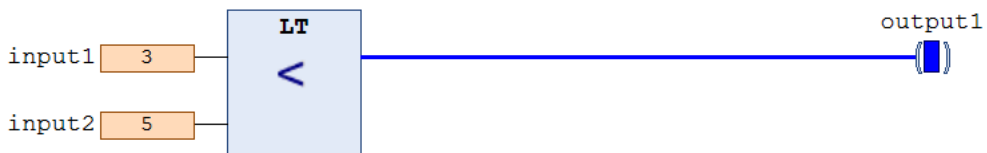
| Scope  | No.      | Type | Description  |
|--------|----------|------|--|
| Input  | (1), (2) | All  | Specifies the variables to be compared.  |
| Output | -        | BOOL | Outputs TRUE if the value of input (1) is less than the value of input (2).<br>Otherwise, outputs FALSE. |

### ■ Program example

This program is designed to compare the input variables “input1” and “input2” and output the result to the output variable “output1”.

#### LD program

TRUE is output because the input variable “input1” is less than the input variable “input2”.



#### ST program

Use the operator (<) to compare the values.

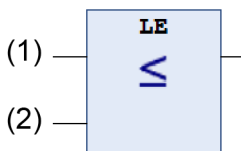
FALSE is output because the input variable “input1” is not less than the input variable “input2”.

```
output1 FALSE := input1 6 < input2 2 ;
```

### 3.4.4 LE (“Less Than or Equal” Comparison)

This is a function that compares two input arguments and determines if the first argument is less than or equal to the second argument.

### ■ Icon



### ■ Parameter

| Scope  | No.      | Type | Description  |
|--------|----------|------|--|
| Input  | (1), (2) | All  | Specifies the variables to be compared.  |
| Output | -        | BOOL | Outputs TRUE if the value of input (1) is less than or equal to the value of input (2).<br>Otherwise, outputs FALSE. |

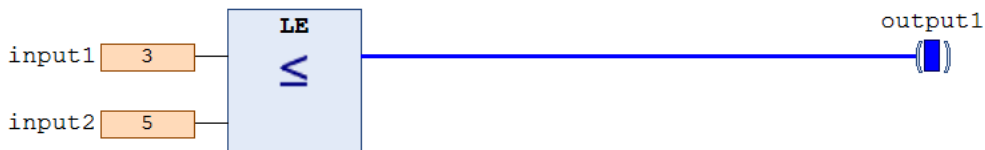
### 3.4 Comparison Operation Instructions

#### ■ Program example

This program is designed to compare the input variables “input1” and “input2” and output the result to the output variable “output1”.

#### LD program

TRUE is output because the input variable “input1” is less than or equal to the input variable “input2”.



#### ST program

Use the operator (<=) to compare the values.

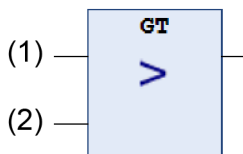
FALSE is output because the input variable “input1” is not less than or equal to the input variable “input2”.

```
output1 FALSE := input1 6 <= input2 2;
```

### 3.4.5 GT (“Greater Than” Comparison)

This is a function that compares two input arguments and determines if the first argument is greater than the second argument.

#### ■ Icon



#### ■ Parameter

| Scope  | No.      | Type | Description   |
|--------|----------|------|---|
| Input  | (1), (2) | All  | Specifies the variables to be compared.   |
| Output | -        | BOOL | Outputs TRUE if the value of input (1) is greater than the value of input (2).<br>Otherwise, outputs FALSE. |

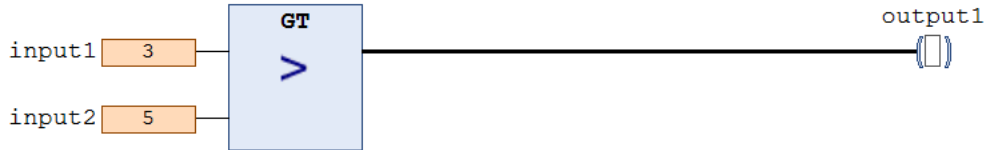
#### ■ Program example

This program is designed to compare the input variables “input1” and “input2” and output the result to the output variable “output1”.



### LD program

FALSE is output because the input variable “input1” is not greater than the input variable “input2”.



### ST program

Use the operator (>) to compare the values.

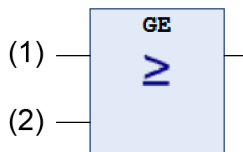
TRUE is output because the input variable “input1” is greater than the input variable “input2”.

```
output1 TRUE := input1 6 > input2 2;
```

### 3.4.6 GE (“Greater Than Or Equal” Comparison)

This is a function that compares two input arguments and determines if the first argument is greater than or equal to the second argument.

#### ■ Icon



#### ■ Parameter

| Scope  | No.      | Type | Description   |
|--------|----------|------|---|
| Input  | (1), (2) | All  | Specifies the variables to be compared.   |
| Output | -        | BOOL | Outputs TRUE if the value of input (1) is greater than or equal to the value of input (2).<br>Otherwise, outputs FALSE. |

#### ■ Program example

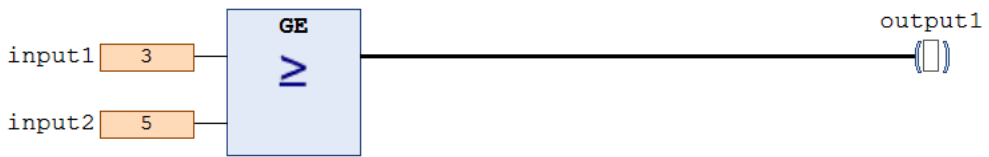
This program is designed to compare the input variables “input1” and “input2” and output the result to the output variable “output1”.

### LD program

FALSE is output because the input variable “input1” is not greater than or equal to the input variable “input2”.

### 3.4 Comparison Operation Instructions

---



#### ST program

Use the operator (>=) to compare the values.

TRUE is output because the input variable "input1" is greater than or equal to the input variable "input2".

```
output1 TRUE := input1 6 >= input2 2 ;
```

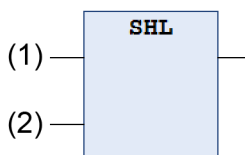
## 3.5 Bit Shift Instructions

Bit shift instructions can be used to perform bit shift operation on input arguments.

### 3.5.1 SHL (Shift Left)

This is a function that shifts the input argument to the left by the specified number of bits and outputs the shifted value. "0" is inserted from the least significant bit up to the bit position shifted by the shift quantity.

#### ■ Icon



#### ■ Parameter

| Scope  | No. | Type     | Description   |
|--------|-----|----------|---|
| Input  | (1) | (Note 1) | Specifies the variable on which bit shift is performed.   |
|        | (2) | (Note 1) | Specifies the number of times bit shift is performed (shift quantity).  |
| Output | -   | (Note 1) | Outputs the value bit shifted to the left from the value of input (1) by the quantity specified in the input (2). |

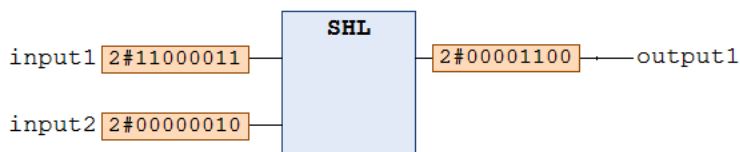
(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT

#### ■ Program example

This program is designed to output the value that is shifted to the left from the value (2#11000011) of input variable "input1" by the number of bits (2 bits) specified in "input2" to the output variable "output1".

#### LD program



#### ST program

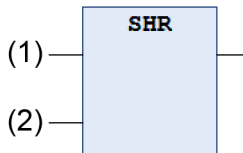
```
output1 [2#00001100] := SHL(input1 [2#11000011], input2 [2#00000010]);
```

## 3.5 Bit Shift Instructions

### 3.5.2 SHR (Shift Right)

This is a function that shifts the input argument to the right by the specified number of bits and outputs the shifted value. “0” is inserted from the most significant bit up to the bit position shifted by the shift quantity.

#### ■ Icon



#### ■ Parameter

| Scope  | No. | Type     | Description  |
|--------|-----|----------|--|
| Input  | (1) | (Note 1) | Specifies the variable on which bit shift is performed.  |
|        | (2) | (Note 1) | Specifies the number of times bit shift is performed (shift quantity).   |
| Output | -   | (Note 1) | Outputs the value bit shifted to the right from the value of input (1) by the quantity specified in the input (2). |

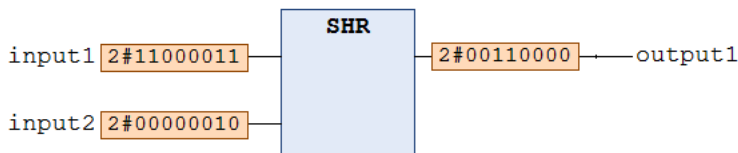
(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT

#### ■ Program example

This program is designed to output the value that is shifted to the right from the value (2#11000011) of input variable “input1” by the number of bits (2 bits) specified in “input2” to the output variable “output1”.

#### LD program



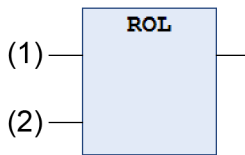
#### ST program

```
output1 2#00110000 := SHR(input1 2#11000011, input2 2#00000010);
```

### 3.5.3 ROL (Rotate Left)

This is a function that shifts the input argument to the left by the specified number of bits and outputs the shifted value. The bit value that has overflowed the most significant bit when the bit is shifted is inserted into the data starting from the least significant bit up to the bit position shifted by the shift quantity.

### ■ Icon



### ■ Parameter

| Scope  | No. | Type     | Description   |
|--------|-----|----------|---|
| Input  | (1) | (Note 1) | Specifies the variable on which bit shift is performed.   |
|        | (2) | (Note 1) | Specifies the number of times bit shift is performed (shift quantity).  |
| Output | -   | (Note 1) | Outputs the value rotated and shifted to the left from the value of input (1) by the quantity specified in the input (2). |

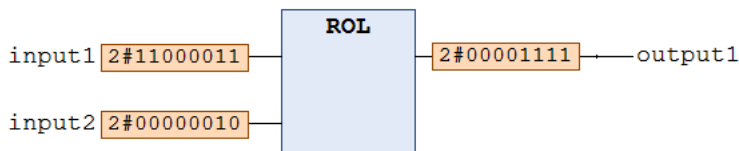
(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT

### ■ Program example

This program is designed to output the value that is rotated and shifted to the left from the value (2#11000011) of input variable “input1” by the number of bits (2 bits) specified in “input2” to the output variable “output1”.

#### LD program



#### ST program

```
output1 2#00001111 := ROL(input1 2#11000011, input2 2#00000010);
```

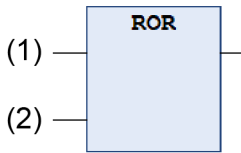
## 3.5.4 ROR (Rotate Right)

This is a function that shifts the input argument to the right by the specified number of bits and outputs the shifted value. The bit value that has overflowed the least significant bit when the bit is shifted is inserted into the data starting from the most significant bit up to the bit position shifted by the shift quantity.

## 3.5 Bit Shift Instructions

---

### ■ Icon



### ■ Parameter

| Scope  | No. | Type     | Description  |
|--------|-----|----------|--|
| Input  | (1) | (Note 1) | Specifies the variable on which bit shift is performed.  |
|        | (2) | (Note 1) | Specifies the number of times bit shift is performed (shift quantity).   |
| Output | -   | (Note 1) | Outputs the value rotated and shifted to the right from the value of input (1) by the quantity specified in the input (2). |

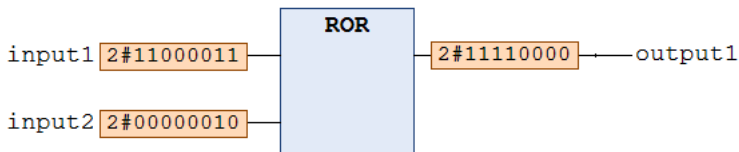
(Note 1) Usable data type

BYTE, WORD, DWORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT

### ■ Program example

This program is designed to output the value that is rotated and shifted to the right from the value (2#11000011) of input variable “input1” by the number of bits (2 bits) specified in “input2” to the output variable “output1”.

#### LD program



#### ST program

```
output1 2#11110000 := ROR(input1 2#11000011, input2 2#00000010);
```

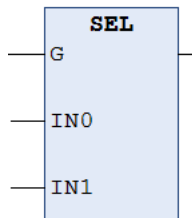
## 3.6 Selection Instructions

Selection instructions can be used to select arguments from multiple input arguments according to specified conditions and to output them.

### 3.6.1 SEL (Binary Selector)

This is a function that outputs the value of the input argument IN0 or IN1 depending on whether the input argument G is true or false.

#### ■ Icon



#### ■ Parameter

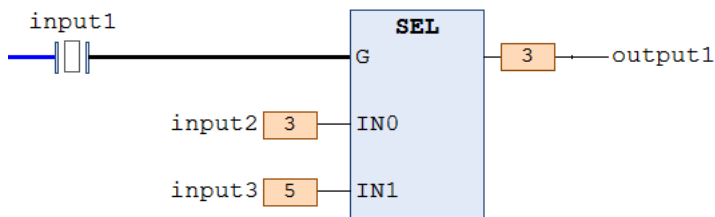
| Scope  | Name | Type | Description  |
|--------|------|------|--|
| Input  | G    | BOOL | Conditions for selecting the contents to be output           |
|        | IN0  | All  | Specifies the variable to be output if G is FALSE.           |
|        | IN1  | All  | Specifies the variable to be output if G is TRUE.            |
| Output | -    | All  | Outputs the value of IN0 or IN1 depending on the value of G. |

#### ■ Program example

This program is designed to output the value of the input variable "input2" or "input3" to the output variable "output1" depending on the value of the input variable "input1".

#### LD program

This program is designed to output the value of "input2" (IN0) because the value of "input1" is FALSE.



#### ST program

This program is designed to output the value of "input3" (IN1) to the "output1" because the value of "input1" is TRUE.

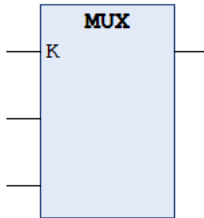
```
output1 [ 5 ] := SEL (input1 [ TRUE ], input2 [ 3 ], input3 [ 5 ] );
```

## 3.6 Selection Instructions

### 3.6.2 MUX (Multiplexer)

This is a function that selectively outputs the input arguments depending on the value of the input argument K.

#### ■ Icon



#### ■ Parameter

| Scope  | Name | Type     | Description   |
|--------|------|----------|---|
| Input  | K    | (Note 1) | Specifies the value (K = 0, 1, 2...) to select the value to output. |
|        | -    | All      | Specifies the value to be output depending on K.                    |
| Output | -    | All      | Outputs one of the input arguments depending on the value of K.     |

(Note 1) Usable data type

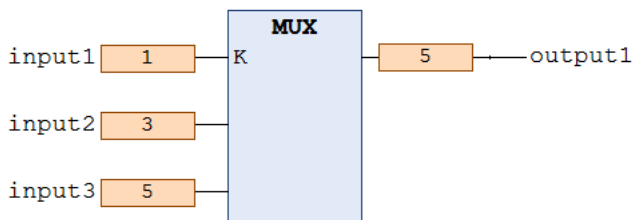
BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT

#### ■ Program example

This program is designed to output the value of the input variable “input2” or “input3” to the output variable “output1” depending on the value of the input variable “input1”.

##### LD program

This program is designed to output the value of “input3” to “output1” depending on the value (1) of “input1”.



##### ST program

This program is designed to output the value of “input 2” to the “output 1” depending on the value (0) of “input 1”.

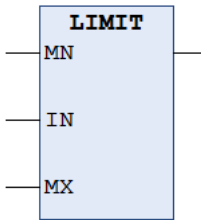
```
output1 [ 3 ] := MUX(input1 [ 0 ], input2 [ 3 ], input3 [ 5 ] );
```



**3.6.3 LIMIT (Limiter)**

This is a function that limits the input value with the lower and upper limit values and outputs a restricted value.

■ **Icon**



■ **Parameter**

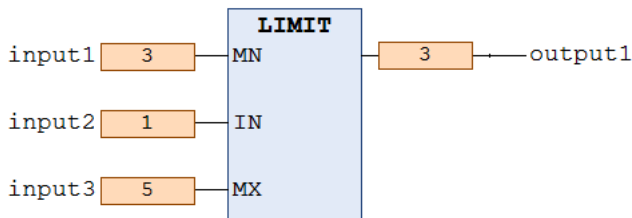
| Scope  | Name | Type | Description   |
|--------|------|------|---|
| Input  | MN   | All  | Specifies the lower limit of the value to be output.  |
|        | IN   | All  | Specifies the input values to be restricted.  |
|        | MX   | All  | Specifies the upper limit of the value to be output.  |
| Output | -    | All  | Outputs values according to the following conditions.<br>$IN \leq MN$ : Outputs "MN".<br>$MN \leq IN \leq MX$ : Outputs "IN".<br>$MX \leq IN$ : Outputs "MX". |

■ **Program example**

This program is designed to limit the value range of the input variable "input2" with the input variable "input1" (lower limit) and the input variable "input3" (upper limit) and to output the limited value to the output variable "output1".

**LD program**

This program is designed to output "3" to "output1" because the value (1) of "input2" (IN) is less than or equal to the lower limit (3) specified in "input1" (MN).



**ST program**

This program is designed to output "5" to "output1" because the value (8) of "input2" is greater than or equal to the upper limit (5) specified in "input3".

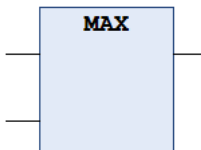
```
output1 [ 5 ] := LIMIT(input1 [ 3 ], input2 [ 8 ], input3 [ 5 ] );
```

## 3.6 Selection Instructions

### 3.6.4 MAX (Maximum Value)

This is a function that outputs the maximum value of the input arguments.

#### ■ Icon



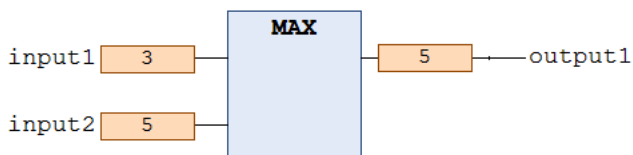
#### ■ Parameter

| Scope  | Type | Description  |
|--------|------|--|
| Input  | All  | Specifies the values from which to obtain the maximum value. |
| Output | All  | Outputs the maximum value of the input values.               |

#### ■ Program example

This program is designed to output the maximum value of the input variables to the output variable "output1".

#### LD program



#### ST program

```
output1 [ 5 ] := MAX(input1 [ 3 ], input2 [ 5 ] );
```

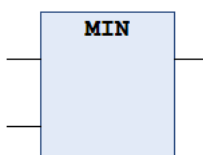
#### **i** Info.

- If you want to increase input arguments in the LD program, right-click on the MAX function, and, on the displayed menu, select "Add Input".

### 3.6.5 MIN (Minimum Value)

This is a function that outputs the minimum value of the input arguments.

### ■ Icon



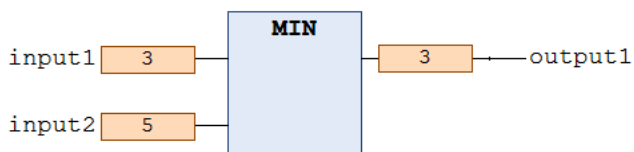
### ■ Parameter

| Scope  | Type | Description  |
|--------|------|--|
| Input  | All  | Specifies the values from which to obtain the minimum value. |
| Output | All  | Outputs the minimum value of the input values.               |

### ■ Program example

This program is designed to output the minimum value of the input variables to the output variable “output1”.

#### LD program



#### ST program

```
output1 [ 3 ] := MIN(input1 [ 3 ], input2 [ 5 ] );
```

### **i** Info.

- If you want to increase input arguments in the LD program, right-click on the MIN function, and, on the displayed menu, select "Add Input".

## 3.7 Numerical Operation Instructions

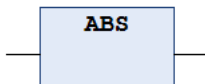
### 3.7 Numerical Operation Instructions

Numerical operation instructions can be used to perform various numerical calculations.

#### 3.7.1 ABS (Absolute Value)

This is a function that outputs the absolute value of the input argument.

##### ■ Icon



##### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the value from which to obtain the absolute value. |
| Output | (Note 1) | Outputs the absolute value of the input argument.            |

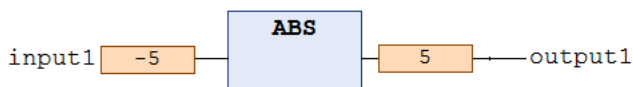
(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

##### ■ Program example

This program is designed to output the absolute value of the input variable "input1" to the output variable "output1".

##### LD program



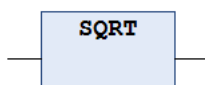
##### ST program

```
output1 5 := ABS(input1 -5);
```

#### 3.7.2 SQRT (Square Root)

This is a function that outputs the square root ( $\sqrt{\quad}$ ) of the input argument.

##### ■ Icon



### ■ Parameter

| Scope  | Type     | Description   |
|--------|----------|---|
| Input  | (Note 1) | Specifies the value from which to obtain the square root. |
| Output | (Note 2) | Outputs the square root of the input argument.            |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

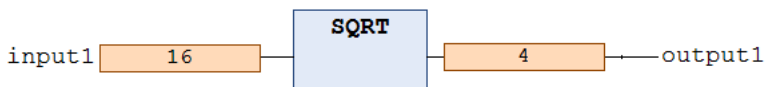
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

### ■ Program example

This program is designed to output the square root of the input variable “input1” to the output variable “output1”.

#### LD program



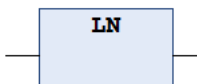
#### ST program

```
output1 4 := SQRT(input1 16);
```

### 3.7.3 LN (Natural Logarithm)

This is a function that outputs the natural logarithm ( $\log_e X$ ) of the input argument ( $X$ ).

### ■ Icon



### ■ Parameter

| Scope  | Type     | Description   |
|--------|----------|---|
| Input  | (Note 1) | Specifies the value from which to obtain the natural logarithm. |
| Output | (Note 2) | Outputs the natural logarithm of the input argument.            |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

(Note 2) Usable data type

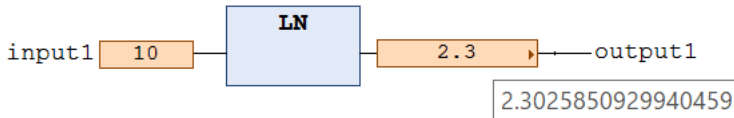
REAL (if the input is REAL), LREAL

## 3.7 Numerical Operation Instructions

### ■ Program example

This program is designed to output the natural logarithm ( $\log_e 10$ ) of the input variable “input1” (10) to the output variable “output1”.

#### LD program



#### ST program

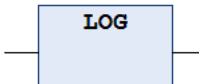
```
output1 2.3 := LN(input1 10);
```

2.3025850929940459

### 3.7.4 LOG (Common Logarithm)

This is a function that outputs the common logarithm ( $\log_{10}X$ ) of the input argument (X).

### ■ Icon



### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the value from which to obtain the common logarithm. |
| Output | (Note 2) | Outputs the common logarithm of the input argument.            |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

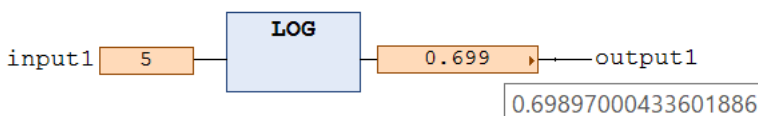
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

### ■ Program example

This program is designed to output the common logarithm ( $\log_{10}5$ ) of the input variable “input1” (5) to the output variable “output1”.

#### LD program



**ST program**

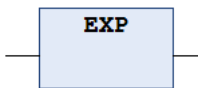
```
output1 0.699 := LOG(input1 5);
```

0.69897000433601886

**3.7.5 EXP (Natural Exponent)**

This is a function that outputs the natural exponent ( $e^X$ ) of the input argument (X).

■ **Icon**



■ **Parameter**

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the value from which to obtain the natural exponent. |
| Output | (Note 2) | Outputs the natural exponent of the input argument.            |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

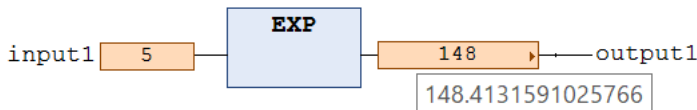
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

■ **Program example**

This program is designed to output the natural exponent of the input variable “input1” to the output variable “output1”.

**LD program**



**ST program**

```
output1 148 := EXP(input1 5);
```

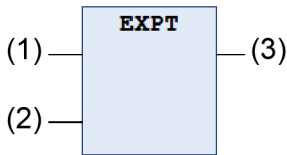
148.4131591025766

**3.7.6 EXPT (Exponentiation)**

This is a function that outputs the exponentiation ( $a^n$ ) of the input arguments (a, n).

## 3.7 Numerical Operation Instructions

### ■ Icon



### ■ Parameter

| Scope  | No. | Type     | Description   |
|--------|-----|----------|---|
| Input  | (1) | (Note 1) | Inputs the base of exponentiation.  |
|        | (2) | (Note 1) | Inputs the exponent of exponentiation.  |
| Output | (3) | (Note 2) | Outputs the exponentiation obtained from the input arguments.<br>Outputs $a^n$ in the following case.<br>Input (1): a<br>Input (2): n |

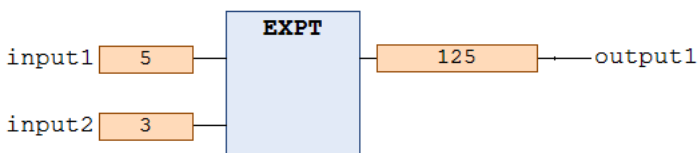
(Note 1) Usable data type  
BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

(Note 2) Usable data type  
REAL (if the input is REAL), LREAL

### ■ Program example

This program is designed to output the exponentiation ( $5^3 = 125$ ) obtained from the input variables "input1" and "input2" to the output variable "output1".

#### LD program



#### ST program

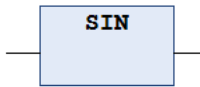
```
output1 [ 125 ] := EXPT(input1 [ 5 ], input2 [ 3 ] );
```

### 3.7.7 SIN (Trigonometric Function Sine)

This is a function that outputs the value of the trigonometric function sine. The unit of the input argument is radian.



### ■ Icon



### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the value (unit: radian) from which to obtain the trigonometric function sine. |
| Output | (Note 2) | Outputs the value of sine of the input argument.   |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

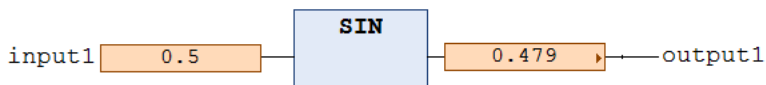
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

### ■ Program example

This program is designed to output the value of the trigonometric function sine obtained from the input variable "input1" to the output variable "output1".

#### LD program



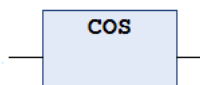
#### ST program

```
output1 [0.479] := SIN(input1 [0.5]);
```

### 3.7.8 COS (Trigonometric Function Cosine)

This is a function that outputs the value of the trigonometric function cosine. The unit of the input argument is radian.

### ■ Icon



## 3.7 Numerical Operation Instructions

### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the value (unit: radian) from which to obtain the trigonometric function cosine. |
| Output | (Note 2) | Outputs the value of cosine of the input argument.   |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

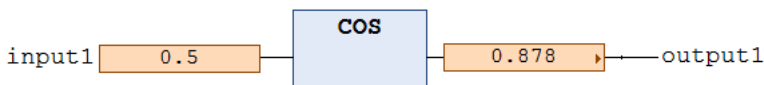
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

### ■ Program example

This program is designed to output the value of the trigonometric function cosine obtained from the input variable "input1" to the output variable "output1".

#### LD program



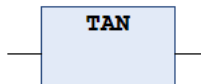
#### ST program

```
output1 0.878 := COS (input1 0.5) ;
```

### 3.7.9 TAN (Trigonometric Function Tangent)

This is a function that outputs the value of the trigonometric function tangent. The unit of the input argument is radian.

### ■ Icon



### ■ Parameter

| Scope  | Type     | Description   |
|--------|----------|---|
| Input  | (Note 1) | Specifies the value (unit: radian) from which to obtain the trigonometric function tangent. |
| Output | (Note 2) | Outputs the value of tangent of the input argument.   |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

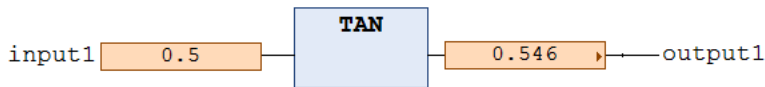
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

### ■ Program example

This program is designed to output the value of the trigonometric function tangent obtained from the input variable "input1" to the output variable "output1".

#### LD program



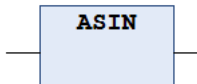
#### ST program

```
output1 0.546 := TAN(input1 0.5);
```

### 3.7.10 ASIN (Trigonometric Function Arc Sine)

This is a function that outputs the value of the trigonometric function arc sine. The unit of the input argument is radian.

### ■ Icon



### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the value (unit: radian) from which to obtain the trigonometric function arc sine. |
| Output | (Note 2) | Outputs the value of arc sine of the input argument.   |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

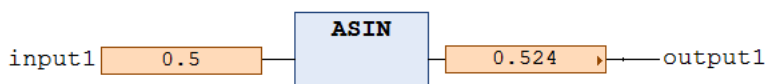
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

### ■ Program example

This program is designed to output the value of the trigonometric function arc sine obtained from the input variable "input1" to the output variable "output1".

#### LD program



## 3.7 Numerical Operation Instructions

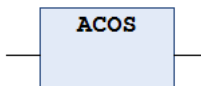
### ST program

```
output1 0.524 := ASIN(input1 0.5);
```

### 3.7.11 ACOS (Trigonometric Function Arc Cosine)

This is a function that outputs the value of the trigonometric function arc cosine. The unit of the input argument is radian.

#### ■ Icon



#### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the value (unit: radian) from which to obtain the trigonometric function arc cosine. |
| Output | (Note 2) | Outputs the value of arc cosine of the input argument.   |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

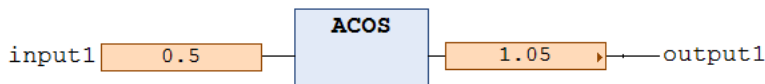
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

#### ■ Program example

This program is designed to output the value of the trigonometric function arc cosine obtained from the input variable "input1" to the output variable "output1".

#### LD program



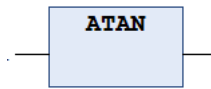
#### ST program

```
output1 1.05 := ACOS(input1 0.5);
```

### 3.7.12 ATAN (Trigonometric Function Arc Tangent)

This is a function that outputs the value of the trigonometric function arc tangent. The unit of the input argument is radian.

### ■ Icon



### ■ Parameter

| Scope  | Type     | Description   |
|--------|----------|---|
| Input  | (Note 1) | Specifies the value (unit: radian) from which to obtain the trigonometric function arc tangent. |
| Output | (Note 2) | Outputs the value of arc tangent of the input argument.   |

(Note 1) Usable data type

BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL

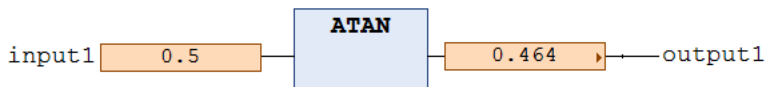
(Note 2) Usable data type

REAL (if the input is REAL), LREAL

### ■ Program example

This program is designed to output the value of the trigonometric function arc tangent obtained from the input variable "input1" to the output variable "output1".

#### LD program



#### ST program

```
output1 [0.464] := ATAN(input1 [0.5]);
```

## 3.8 Data Type Conversion Instructions

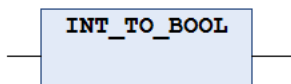
### 3.8 Data Type Conversion Instructions

Data type conversion instructions can be used to convert the data type of a variable.

#### 3.8.1 "Type 1"\_TO\_"Type 2" ("Type 1">"Type 2" Conversion)

This is a function that converts the data type of the input argument "Type 1" to another data type "Type 2". Conversion from a larger size data type to a smaller size data type is not performed automatically. It is necessary to convert the data type using this instruction.

##### ■ Icon



##### ■ Parameter

| Scope  | Type | Description  |
|--------|------|--|
| Input  | All  | Specifies the variable required to be converted to a different type. |
| Output | All  | Outputs the variable converted to a different type.                  |

##### ■ Types of functions and conversion examples

| Types of functions | Conversion example |               | Description   |
|--------------------|--------------------|---------------|---|
|                    | Input              | Output        |   |
| INT_TO_BOOL        | 5                  | TRUE          | If other than 0, outputs TRUE.                                    |
| UINT_TO_USINT      | 300<br>(16#012C)   | 44<br>(16#2C) | Outputs lower eight bits out of the 16 bits of UINT.              |
| REAL_TO_INT        | 3.5                | 4             | Outputs data after rounding decimals to the nearest whole number. |
| TIME_TO_STRING     | t#2s5ms            | 'T#2s5ms'     | Outputs the TIME constant in the STRING type.                     |
| TOD_TO_UDINT       | tod#00:30:20       | 182000        | Outputs the milliseconds from 00:00:00.                           |
| DATE_TO_UDINT      | d#2000-1-1         | 946684800     | Outputs the seconds from 1970-1-1.                                |
| STRING_TO_WORD     | '123'              | 123           | Outputs the character string of a numerical value.                |

##### ■ Program example

###### LD program

The following is a program example of the INT\_TO\_BOOL function.



###### ST program

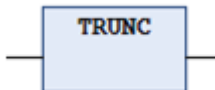
The following is a program example of the TOD\_TO\_UDINT function.

```
output1 1820000 := TOD_TO_UDINT(input1 TOD#0:30:20);
```

### 3.8.2 TRUNC (Real Number to DINT Conversion)

This is a function that converts a real number type input to a DINT type.

■ **Icon**



■ **Parameter**

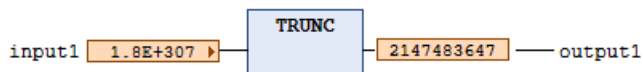
| Scope  | Type           | Description   |
|--------|----------------|---|
| Input  | REAL,<br>LREAL | Real number type value  |
| Output | DINT           | Outputs the value converted to the DINT type from the input argument. |

■ **Program example**

This program is designed to convert the LREAL type input variable “input1” to the DINT type output variable “output1” and output the converted data.

```
Input1 := 1.7976931348623157E+307;
```

**LD program**



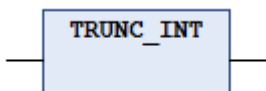
**ST program**

```
output1 2147483647 := TRUNC(input1 1.8E+307);
```

### 3.8.3 TRUNC\_INT (Real Number to INT Conversion)

This is a function that converts a real number type input to an INT type.

■ **Icon**



## 3.8 Data Type Conversion Instructions

---

### ■ Parameter

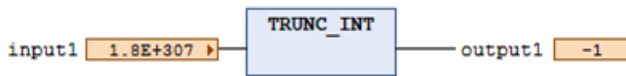
| Scope  | Type           | Description  |
|--------|----------------|--|
| Input  | REAL,<br>LREAL | Real number type value   |
| Output | INT            | Outputs the value converted to the INT type from the input argument. |

### ■ Program example

This program is designed to convert the LREAL type input variable “input1” to the INT type output variable “output1” and output the converted data.

Input1 := 1.7976931348623157E+307;

#### LD program



#### ST program

```
output1[-1] := TRUNC_INT(input1[1.8E+307]);
```



### 3.9 Character string instructions

Character string instructions can be used to perform various operations on character strings.

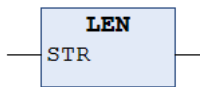
There is no limit to the length of a STRING type string, but the string functions described in this chapter only process lengths of 1 to 255 characters.

Do not use a string longer than 256 characters in the function input.

#### 3.9.1 LEN (Length of a Character String)

This is a function that outputs the length of a character string.

##### ■ Icon



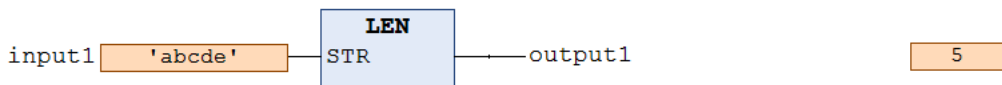
##### ■ Parameter

| Scope  | Name | Type   | Description   |
|--------|------|--------|---|
| Input  | STR  | STRING | Specifies the character string from which to obtain the length. |
| Output | -    | INT    | Outputs the character string length of the input argument.      |

##### ■ Program example

This program is designed to output the character string length of the input variable "input1" to the output variable "output1".

##### LD program



##### ST program

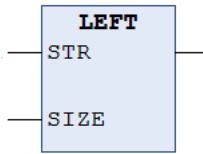
```
output1 [ 5 ] := LEN(input1 [ 'abcde' ] );
```

#### 3.9.2 LEFT (Extracting Characters from the Left End)

This is a function that extracts a character string consisting of the specified number of characters from the left end of the character string and outputs the extracted data.

## 3.9 Character string instructions

### ■ Icon



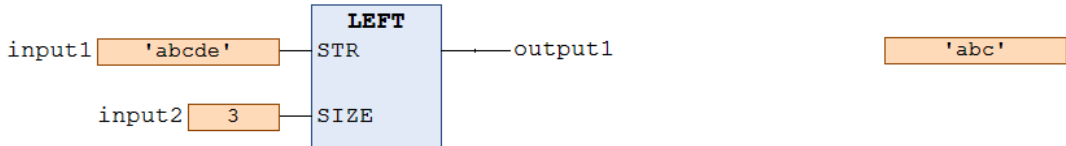
### ■ Parameter

| Scope  | Name | Type   | Description   |
|--------|------|--------|---|
| Input  | STR  | STRING | Specifies the character string from which a character string is to be extracted.  |
|        | SIZE | INT    | Specifies the number of characters to be extracted from the left.   |
| Output | -    | STRING | Extracts a character string consisting of the number of characters specified in SIZE from STR and outputs the extracted data. |

### ■ Program example

This program is designed to extract a character string consisting of the number of characters (3 characters) specified in “input2” from the left end of the character string of the input variable “input1” and to output the extracted character string to the output variable “output1”.

#### LD program



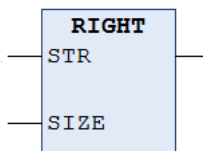
#### ST program

```
output1 := LEFT(input1, input2);
```

### 3.9.3 RIGHT (Extracting Characters from the Right End)

This is a function that extracts a character string consisting of the specified number of characters from the right end of the character string and outputs the extracted data.

### ■ Icon



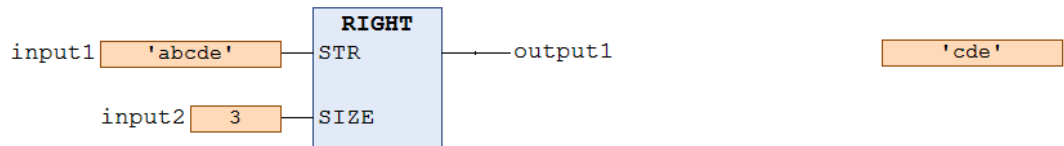
### ■ Parameter

| Scope  | Name | Type   | Description   |
|--------|------|--------|---|
| Input  | STR  | STRING | Specifies the character string from which a character string is to be extracted.  |
|        | SIZE | INT    | Specifies the number of characters to be extracted from the right.  |
| Output | -    | STRING | Extracts a character string consisting of the number of characters specified in SIZE from STR and outputs the extracted data. |

### ■ Program example

This program is designed to extract a character string consisting of the number of characters (3 characters) specified in “input2” from the right end of the character string of the input variable “input1” and to output the extracted character string to the output variable “output1”.

#### LD program



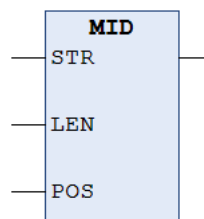
#### ST program

```
output1 'cde' := RIGHT(input1 'abcde', input2 3);
```

### 3.9.4 MID (Extracting Characters from the Specified Position)

This is a function that extracts a character string consisting of the specified number of characters from the specified position of the character string and outputs the extracted data.

### ■ Icon



### ■ Parameter

| Scope | Name | Type   | Description  |
|-------|------|--------|--|
| Input | STR  | STRING | Specifies the character string from which a character string is to be extracted. |
|       | LEN  | INT    | Specifies the number of characters to be extracted.                              |
|       | POS  | INT    | Specified the position from which extraction is to be started.                   |

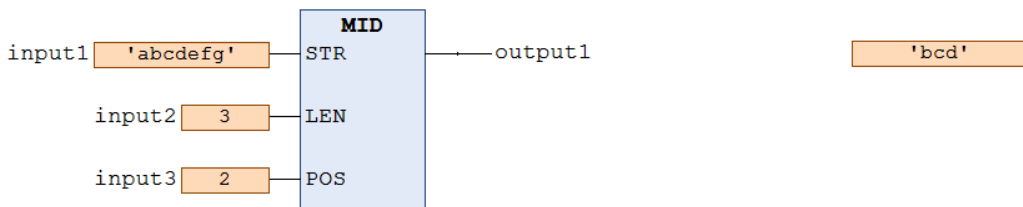
### 3.9 Character string instructions

| Scope  | Name | Type   | Description  |
|--------|------|--------|--|
| Output | -    | STRING | Extracts a character string consisting of the number of characters specified in LEN from STR starting from the position specified in POS and outputs the extracted data. |

#### ■ Program example

This program is designed to extract a character string consisting of the number of characters (3 characters) specified in “input2” from the character string of the input variable “input1”, starting from the position (2nd character from the left end) specified in “input3”, and to output the extracted data to the output variable “output1”.

#### LD program



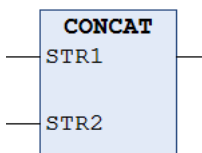
#### ST program

```
output1 'bcd' := MID(input1 'abcdefg', input2 3, input3 2);
```

### 3.9.5 CONCAT (Concatenating Character Strings)

This is a function that concatenates the character strings.

#### ■ Icon



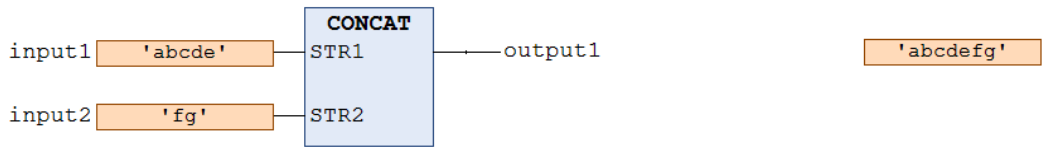
#### ■ Parameter

| Scope  | Name | Type   | Description   |
|--------|------|--------|---|
| Input  | STR1 | STRING | Specifies the character string to be concatenated.  |
|        | STR2 | STRING | Specifies the character string to be concatenated.  |
| Output | -    | STRING | Concatenate the STR2 character string to the right of the STR1 character string and output the concatenated data. |

#### ■ Program example

This program is designed to concatenate the character string of “input2” to the character string of the input variable “input1” and to output the concatenated data to the output variable “output1”.

**LD program**



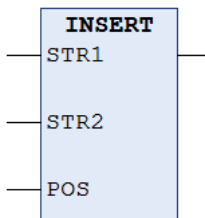
**ST program**

```
output1 := CONCAT(input1, input2);
```

**3.9.6 INSERT (Inserting a Character String)**

This is a function that inserts a character string in the specified position and outputs the inserted data.

■ **Icon**



■ **Parameter**

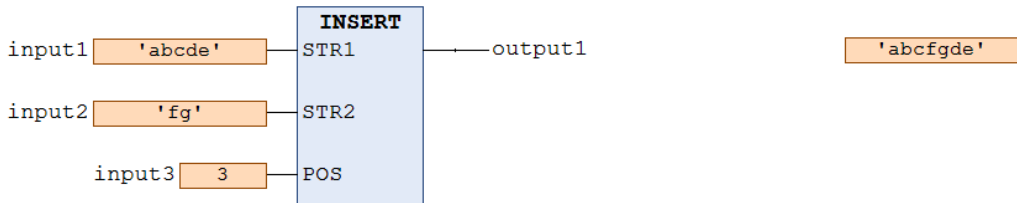
| Scope  | Name | Type   | Description  |
|--------|------|--------|--|
| Input  | STR1 | STRING | Specifies the character string in which a character string is to be inserted.                    |
|        | STR2 | STRING | Specifies the character string to be inserted.   |
|        | POS  | INT    | Specifies the position to be inserted.<br>n-th character from the left                           |
| Output | -    | STRING | Inserts the STR2 character string in the position specified in POS in the STR1 character string. |

■ **Program example**

This program is designed to insert the character string of “input2” in the position (3rd character from the left end) specified in “input3” from the left of the the character string of the input variable “input1” and to output the inserted data to the output variable “output1”.

## 3.9 Character string instructions

### LD program



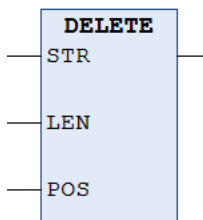
### ST program

```
output1["'abcfgde'"] := INSERT(input1["'abcde'"], input2["'fg'"], input3["3"]);
```

### 3.9.7 DELETE (Deleting a Character String)

This is a function that deletes a character string from the specified position and outputs the deleted data.

#### ■ Icon



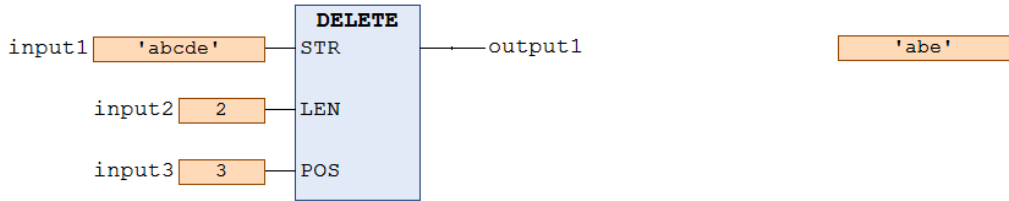
#### ■ Parameter

| Scope  | Name | Type   | Description  |
|--------|------|--------|--|
| Input  | STR  | STRING | Specifies the character string from which a character string is to be deleted.   |
|        | LEN  | INT    | Specifies the length of the character string to be deleted.  |
|        | POS  | INT    | Specified the position from which deletion is to be started. n-th character from the left  |
| Output | -    | STRING | Deletes a character string consisting of the number of characters specified in LEN from the left end of the STR character string starting from the position specified in POS and outputs the deleted data. |

#### ■ Program example

This program is designed to delete a character string consisting of the number of characters (2 characters) specified in “input2” from the character string of the input variable “input1” starting from the position (3rd character from the left) specified in “input3” and to output the deleted data to the output variable “output1”.

**LD program**



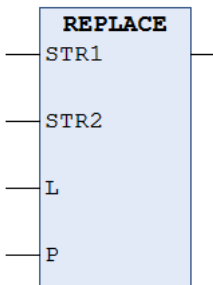
**ST program**

```
output1 := DELETE(input1, input2, input3);
```

**3.9.8 REPLACE (Replacing a Character String)**

This is a function that replaces the character strings and outputs the replaced character strings.

■ **Icon**



■ **Parameter**

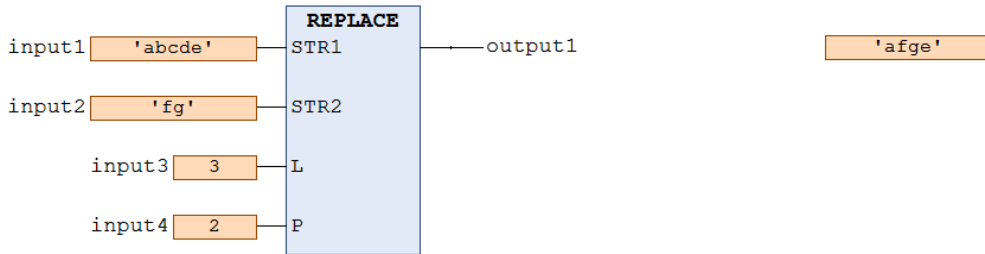
| Scope  | Name | Type   | Description   |
|--------|------|--------|---|
| Input  | STR1 | STRING | Specifies the character string to be replaced.  |
|        | STR2 | STRING | Specifies the character string to be added by replacement.  |
|        | L    | INT    | Specifies the number of characters to be deleted by replacement.  |
|        | P    | INT    | Specifies the position where the character string specified by STR2 is to be added by replacement.  |
| Output | -    | STRING | Replaces the number of characters specified in L with the character string specified in STR2 from the left end of the character string specified in STR1 starting from the position specified in P and outputs the replaced data. |

■ **Program example**

This program is designed to replace a character string consisting of the number of characters specified in “input3” with the character string specified in “input2” from the position specified in “input4” in the character string of the input variable “input1” and to output the replaced data to the output variable “output1”.

### 3.9 Character string instructions

#### LD program



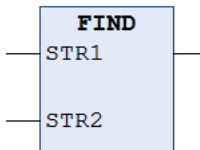
#### ST program

```
output1 'afge' := REPLACE(input1 'abcde', input2 'fg',
                          input3 3, input4 2);
```

### 3.9.9 FIND (Search for Character String)

This is a function that searches for a specified character string and outputs the searched position.

#### ■ Icon



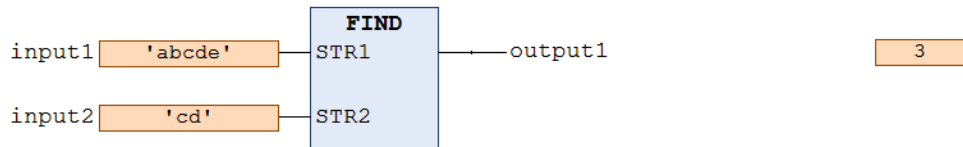
#### ■ Parameter

| Scope  | Name | Type   | Description   |
|--------|------|--------|---|
| Input  | STR1 | STRING | Specifies the character string in which a character string is to be searched for.   |
|        | STR2 | STRING | Specifies s character string to search for in the character string specified in STR1.   |
| Output | -    | INT    | Searches for the character string specified in STR2 in the character string specified in STR1 and outputs the position from the left end. |

#### ■ Program example

This program is designed to search for the character string specified in “input2” in the character string of the input variable “input1” and to output the position from the left to the output variable “output1”.



**LD program****ST program**

```
output1 3 := FIND(input1 'abcde', input2 'cd');
```

**i Info.**

- Outputs 0 if the character string is not found.
- If the character string is found in multiple places, the position found first (the leftmost position) is output.

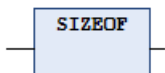
## 3.10 Other Instructions

### 3.10 Other Instructions

#### 3.10.1 SIZEOF (Get the Size)

This is a function that outputs the size (number of bytes) of the input argument.

##### ■ Icon



##### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Specifies the argument whose size is to be calculated. |
| Output | (Note 1) | Outputs the size of (1).                               |

(Note 1) Usable data types

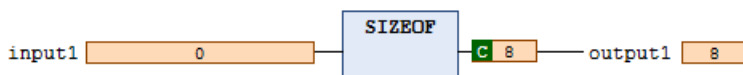
All standard data types

(BOOL, BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL, TIME, LTIME, DATE, TIME\_OF\_DAY, DATE\_AND\_TIME, STRING, WSTRING)

##### ■ Program example

This program is designed to output the size of the ULINT type input variable “input1” to the UINT type output variable “output1”.

##### LD program



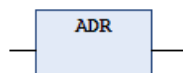
##### ST program

```
output1 8 := SIZEOF(input1 0);
```

#### 3.10.2 ADR (Get the Address)

This is a function that outputs the address of the variable.

##### ■ Icon



### ■ Parameter

| Scope  | Type     | Description  |
|--------|----------|--|
| Input  | (Note 1) | Input the variable from which to get the address.    |
| Output | (Note 1) | Outputs the address (pointer) of the input variable. |

(Note 1) Usable data types

All standard data types

(BOOL, BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL, TIME, LTIME, DATE, TIME\_OF\_DAY, DATE\_AND\_TIME, STRING, WSTRING)

### ■ Usable data type

All standard data types

(BOOL, BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, LREAL, TIME, LTIME, DATE, TIME\_OF\_DAY, DATE\_AND\_TIME, STRING, WSTRING)

### ■ Program example

This program is designed to output the address of the input variable “input1” to the output variable “output1”.

#### LD program



#### ST program

```
output1[16#F1D61884] := ADR(input1[0]);
```

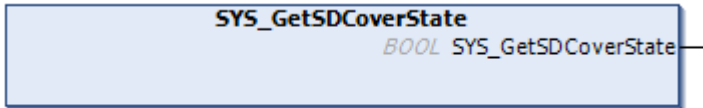
## 3.11 SD Memory Card Slot Instruction

### 3.11 SD Memory Card Slot Instruction

#### 3.11.1 SYS\_GetSDCoverState (Get SD Card Cover Open / Close State)

This is a function that gets an open / close state of the card cover for the SD memory card slot.

##### ■ Icon



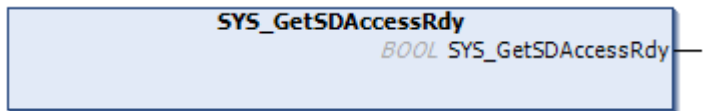
##### ■ Parameter

| Scope  | Name                | Type | Description   |
|--------|---------------------|------|---|
| Output | SYS_GetSDCoverState | BOOL | TRUE: The card cover is closed.<br>FALSE: The card cover is open. |

#### 3.11.2 SYS\_GetSDAccessRdy (Get SD Card Access Ready State)

This is a function block that gets the state whether an access to the SD memory card is allowed.

##### ■ Icon



##### ■ Parameter

| Scope  | Name               | Type | Description  |
|--------|--------------------|------|--|
| Output | SYS_GetSDAccessRdy | BOOL | TRUE: Access to the SD memory card is enabled.<br>FALSE: Access to the SD memory card is disabled. |

# 4 Function Blocks (Basic Instructions)

---

|       |  |      |
|-------|--|------|
| 4.1   | Timer Instructions .....                   | 4-2  |
| 4.1.1 | TON (Timer ON).....                        | 4-2  |
| 4.1.2 | TOF (Timer OFF) .....                      | 4-3  |
| 4.1.3 | TP (Timer Pulse).....                      | 4-4  |
| 4.1.4 | RTC (Realtime Clock).....                  | 4-6  |
| 4.2   | Counter Instructions.....                  | 4-7  |
| 4.2.1 | CTU (Up Counter).....                      | 4-7  |
| 4.2.2 | CTD (Down Counter) .....                   | 4-8  |
| 4.2.3 | CTUD (Up-down Counter) .....               | 4-9  |
| 4.3   | Edge Detection Instructions .....          | 4-11 |
| 4.3.1 | R_TRIG (Rising Edge Detection).....        | 4-11 |
| 4.3.2 | F_TRIG (Falling Edge Detection).....       | 4-11 |
| 4.4   | Bistable Circuit Instructions.....         | 4-13 |
| 4.4.1 | SR (Set-priority Bistable Circuit) .....   | 4-13 |
| 4.4.2 | RS (Reset-priority Bistable Circuit) ..... | 4-14 |

## 4.1 Timer Instructions

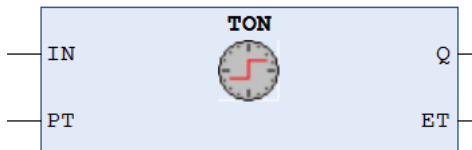
### 4.1 Timer Instructions

Timer instructions can be used to perform timer operations.

#### 4.1.1 TON (Timer ON)

This is a function block (FB) that starts the timer when the input becomes TRUE. After a specified time elapses, the output becomes TRUE.

##### ■ Icon



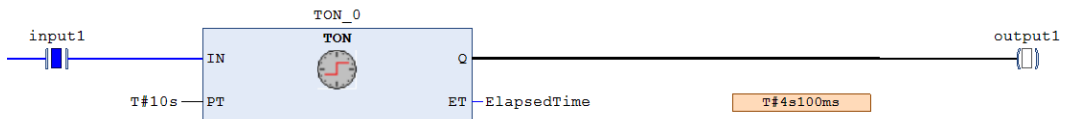
##### ■ Parameter

| Scope  | Name | Type | Description   |
|--------|------|------|---|
| Input  | IN   | BOOL | Starts the timer when FALSE becomes TRUE and the timer continues counting while it remains TRUE.<br>Resets the timer when it becomes FALSE. |
|        | PT   | TIME | Specifies the timer time.   |
| Output | Q    | BOOL | Outputs TRUE when the time specified in the input argument PT elapses.  |
|        | ET   | TIME | Specifies the elapsed time of the timer.  |

##### ■ Program example

This program is designed to start the timer when the input variable “input1” becomes TRUE and, after an elapse of 10 seconds, to cause the output variable “output1” to become TRUE. The instance name is TON\_0.

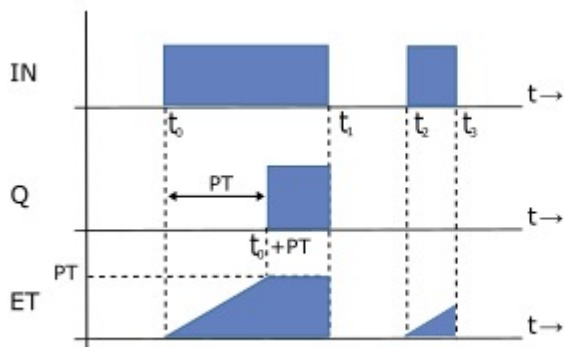
##### LD program



##### ST program

```
TON_0 (IN TRUE := input1 TRUE, PT T#10s := T#10s,
      Q FALSE => output1 FALSE, ET T#3s659ms => ElapsedTime T#3s659ms);
```

### ■ Time-sequence diagram



#### 4.1.2 TOF (Timer OFF)

This is a function block (FB) that starts the timer when the input becomes FALSE. After a specified time elapses, the output becomes FALSE.

### ■ Icon



### ■ Parameter

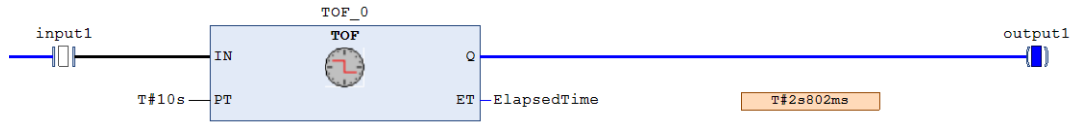
| Scope  | Name | Type | Description   |
|--------|------|------|---|
| Input  | IN   | BOOL | Starts the timer when TRUE becomes FALSE and the timer continues counting while it remains FALSE.<br>Resets the timer when it becomes TRUE. |
|        | PT   | TIME | Specifies the timer time.   |
| Output | Q    | BOOL | Outputs FALSE when the time specified in the input argument PT elapses.   |
|        | ET   | TIME | Specifies the elapsed time of the timer.  |

### ■ Program example

This program is designed to start the timer when the input variable “input1” changes from TRUE to FALSE and, after an elapse of 10 seconds, to cause the output variable “output1” to become FALSE. The instance name is TOF\_0.

## 4.1 Timer Instructions

### LD program

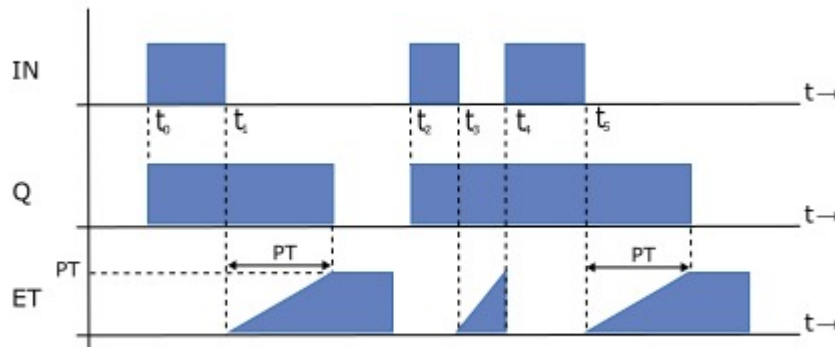


### ST program

```

TOF_0(IN FALSE := input1 FALSE,
      PT T#10s := T#10s,
      Q TRUE => output1 TRUE,
      ET T#2s113ms => ElapsedTime T#2s113ms);
  
```

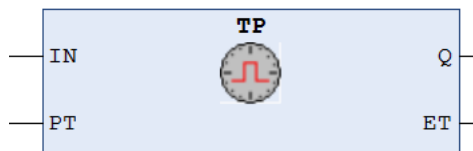
### ■ Time-sequence diagram



### 4.1.3 TP (Timer Pulse)

This is a function block that starts the timer at the rising edge. The output remains TRUE while the timer keeps counting. After a specified time elapses, the output becomes FALSE.

#### ■ Icon



#### ■ Parameter

| Scope | Name | Type | Description  |
|-------|------|------|--|
| Input | IN   | BOOL | Starts the timer when FALSE changes to TRUE (rising edge). |



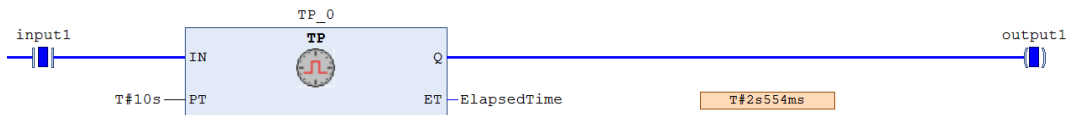
| Scope  | Name | Type | Description  |
|--------|------|------|--|
|        |      |      | Resets the timer when the timer expires and TRUE changes to FALSE .  |
|        | PT   | TIME | Specifies the timer time.  |
| Output | Q    | BOOL | Outputs TRUE from when the timer is started until when the time specified in the input argument PT elapses.<br>Outputs FALSE after the specified time elapses. |
|        | ET   | TIME | Specifies the elapsed time of the timer.   |

■ Program example

This program is designed to start the timer when the input variable “input1” changes from FALSE to TRUE and, during the time from when the timer is started to when the timer expires (for 10 seconds), to cause the output variable “output1” to remain TRUE.

The instance name is TP\_0.

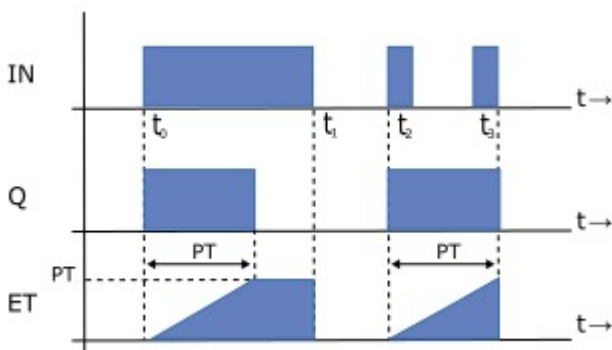
LD program



ST program

```
TP_0 (IN TRUE := input1 TRUE ,
      PT T#10s := T#10s ,
      Q TRUE => output1 TRUE ,
      ET T#2s822ms => ElapsedTime T#2s822ms ) ;
```

■ Time-sequence diagram

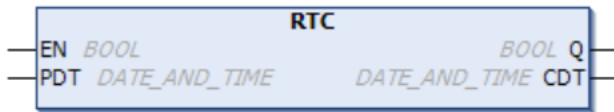


## 4.1 Timer Instructions

### 4.1.4 RTC (Realtime Clock)

This is a function block that starts counting time at the rising edge starting from the specified date and time. The output remains TRUE while the time counting continues. After a specified time elapses, the output becomes FALSE.

#### ■ Icon



#### ■ Parameter

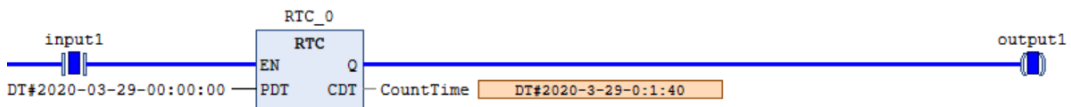
| Scope  | Name | Type          | Description  |
|--------|------|---------------|--|
| Input  | EN   | BOOL          | Starts counting time from the date and time specified in the input argument PDT when FALSE changes to TRUE (rising edge).<br>When TRUE changes to FALSE, DT#1970-01-01-00:00:00 is set in the output argument CDT. |
|        | PDT  | DATE_AND_TIME | Date and time when time counting starts  |
| Output | Q    | BOOL          | Outputs TRUE while time counting continues.  |
|        | CDT  | DATE_AND_TIME | Outputs the time count time from the date and time specified in the input argument PDT.  |

#### ■ Program example

This program is designed to start counting time, starting from 0 o'clock of March 29, 2020, when the input variable "input1" changes from FALSE to TRUE, and, to cause the output variable "output1" to remain TRUE while time counting continues.

The instance name is RTC\_0.

#### LD program



#### ST program

```

RTC_0 (EN TRUE := input1 TRUE ,
      PDT DT#2020-3-29-0:0:0 := DT#2020-03-29-00:00:00 ,
      Q TRUE => output1 TRUE ,
      CDT DT#2020-3-29-0:1:31 => CountTime DT#2020-3-29-0:1:31 ) ;

```

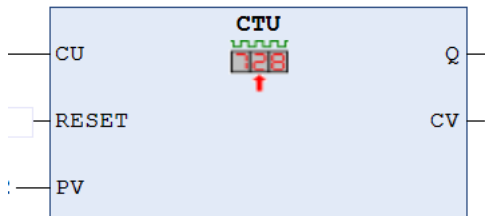
## 4.2 Counter Instructions

Counter instructions can be used to perform counter operations.

### 4.2.1 CTU (Up Counter)

This is a function block that increments the counter value by 1 every time the rising edge occurs.

#### ■ Icon



#### ■ Parameter

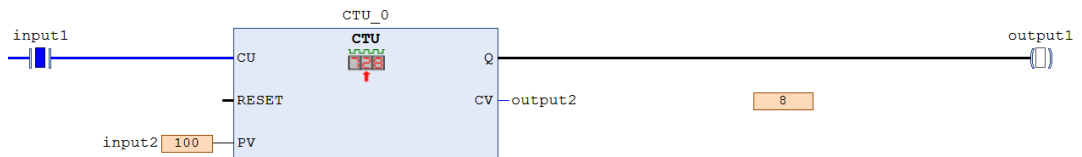
| Scope  | Name  | Type | Description  |
|--------|-------|------|--|
| Input  | CU    | BOOL | Increases the value of the output argument CV by 1 when FALSE changes to TRUE (rising edge). |
|        | RESET | BOOL | If TRUE, 0 is set in the output argument CV.   |
|        | PV    | WORD | Target value of CV   |
| Output | Q     | BOOL | Outputs TRUE when the CV value reaches the PV value.   |
|        | CV    | WORD | Outputs the current counter value.   |

#### ■ Program example

This program is designed to increment the value of the output variable “output2” by 1 every time the input variable “input1” changes from FALSE to TRUE. The program is designed to cause the output variable “output1” to change to TRUE when the value (100) of the input variable “input2” is counted up.

The instance name is CTU\_0.

#### LD program



## 4.2 Counter Instructions

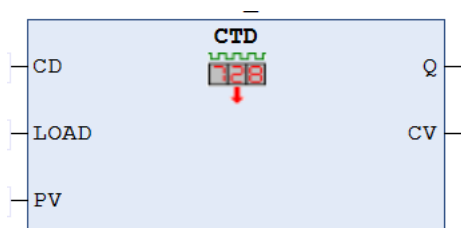
### ST program

```
CTU_0(  
  CU TRUE := input1 TRUE ,  
  RESET:= ,  
  PV 100 := 100 ,  
  Q FALSE => output1 FALSE ,  
  CV 8 => output2 8 );
```

### 4.2.2 CTD (Down Counter)

This is a function block that decrements the counter value by 1 every time the rising edge occurs.

#### ■ Icon



#### ■ Parameter

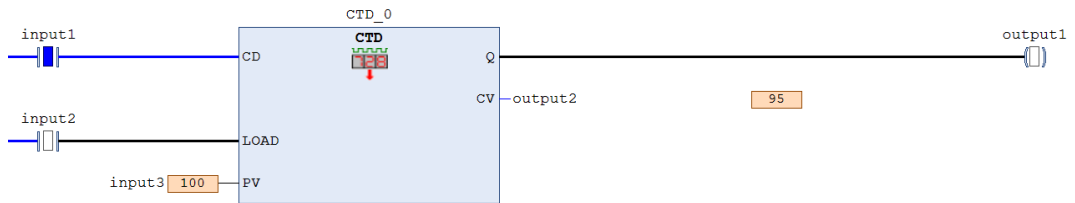
| Scope  | Name | Type | Description   |
|--------|------|------|---|
| Input  | CD   | BOOL | Decrements the value of the output argument CV by 1 when FALSE changes to TRUE (rising edge). |
|        | LOAD | BOOL | If TRUE, the value specified in PV is set in the output argument CV.                          |
|        | PV   | WORD | Initial value of the counter value  |
| Output | Q    | BOOL | Outputs TRUE when the CV value becomes 0.   |
|        | CV   | WORD | Outputs the current counter value.  |

#### ■ Program example

This program is designed to decrement the value of the output variable “output2” by 1 every time the input variable “input1” changes from FALSE to TRUE, and to cause the output variable “output1” to change to TRUE when the value becomes 0. The initial value (100) to count down from is specified in the input variable “input3”.

The instance name is CTD\_0.

**LD program**



**ST program**

```

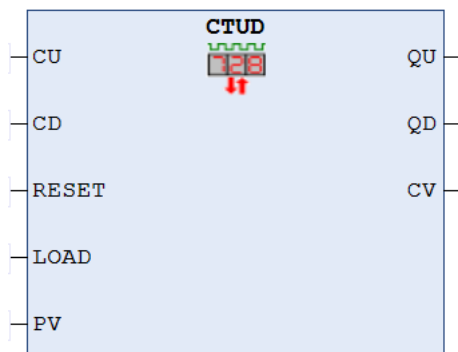
CTD_0 (
  CD TRUE := input1 TRUE ,
  LOAD FALSE := input2 FALSE ,
  PV 100 := input3 100 ,
  Q FALSE => output1 FALSE ,
  CV 95 => output2 95 ) ;

```

**4.2.3 CTUD (Up-down Counter)**

This is a function block that increments or decrements the counter value by 1 every time the rising edge occurs.

■ **Icon**



■ **Parameter**

| Scope  | Name  | Type | Description   |
|--------|-------|------|---|
| Input  | CU    | BOOL | Increments the value of the output argument CV by 1 when FALSE changes to TRUE (rising edge). |
|        | CD    | BOOL | Decrements the value of the output argument CV by 1 when FALSE changes to TRUE (rising edge). |
|        | RESET | BOOL | If TRUE, 0 is set in the output argument CV.  |
|        | LOAD  | BOOL | If TRUE, the value specified in PV is set in the output argument CV.                          |
|        | PV    | WORD | Initial value of the counter value  |
| Output | QU    | BOOL | Outputs TRUE when the CV value reaches the PV value.  |

## 4.2 Counter Instructions

| Scope | Name | Type | Description                               |
|-------|------|------|---|
|       | QD   | BOOL | Outputs TRUE when the CV value becomes 0. |
|       | CV   | WORD | Outputs the current counter value.        |

### ■ Program example

Every time the input variable “input1” changes from FALSE to TRUE, the value of the output variable “output3” is incremented by 1.

Every time the input variable “input2” changes from FALSE to TRUE, the value of the output variable “output3” is decremented by 1

When the output variable “output3” becomes greater than or equal to the input variable “input5”, the output variable “output1” becomes TRUE.

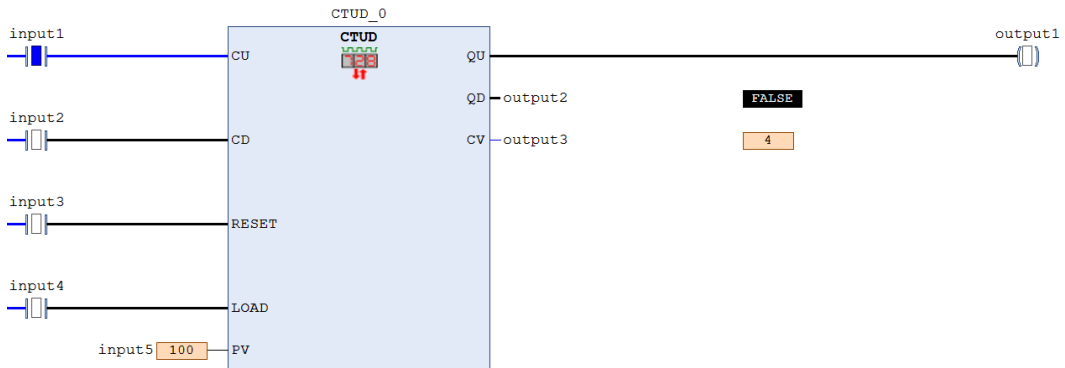
When the output variable “output3” becomes 0, the output variable “output2” becomes TRUE.

When the input variable “input3” becomes TRUE, the output variable “output3” becomes 0.

When the input variable “input4” becomes TRUE, the value (100) of the input variable “input5” is set in the output variable “output3”.

The instance name is CTUD\_0.

### LD program



### ST program

```
CTUD_0 (
  CU TRUE := input1 TRUE ,
  CD FALSE := input2 FALSE ,
  RESET FALSE := input3 FALSE ,
  LOAD FALSE := input4 FALSE ,
  PV 100 := input5 100 ,
  QU FALSE => output1 FALSE ,
  QD FALSE => output2 FALSE ,
  CV 3 => output3 3 );
```

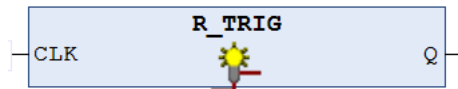
## 4.3 Edge Detection Instructions

Edge detection instructions can be used to perform edge detection.

### 4.3.1 R\_TRIG (Rising Edge Detection)

This is a function block that detects a rising edge.

#### ■ Icon



#### ■ Parameter

| Scope  | Name | Type | Description   |
|--------|------|------|---|
| Input  | CLK  | BOOL | Input that detects a rising edge  |
| Output | Q    | BOOL | Outputs TRUE for one cycle only when a rising edge is detected in the input argument CLK. |

#### ■ Program example

When the input variable “input1” changes from FALSE to TRUE, the output variable “output1” becomes TRUE for one cycle only.

The instance name is R\_TRIG\_0.

#### LD program



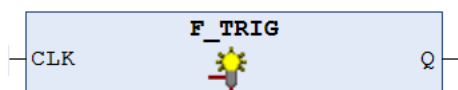
#### ST program

```
R_TRIG_0 (
  CLK TRUE := input1 TRUE ,
  Q TRUE => output1 TRUE );
```

### 4.3.2 F\_TRIG (Falling Edge Detection)

This is a function block that detects a falling edge.

#### ■ Icon



## 4.3 Edge Detection Instructions

---

### ■ Parameter

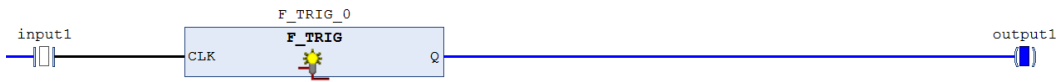
| Scope  | Name | Type | Description  |
|--------|------|------|--|
| Input  | CLK  | BOOL | Input that detects a falling edge  |
| Output | Q    | BOOL | Outputs TRUE for one cycle only when a falling edge is detected in the input argument CLK. |

### ■ Program example

When the input variable “input1” changes from FALSE to TRUE, the output variable “output1” becomes TRUE for one cycle only.

The instance name is F\_TRIG\_0.

#### LD program



#### ST program

```
F_TRIG_0(  
  CLK FALSE := input1 FALSE,  
  Q TRUE => output1 TRUE );
```



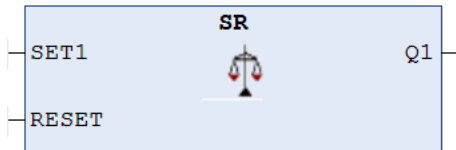
## 4.4 Bistable Circuit Instructions

Bistable circuit instructions can be used to perform edge detection.

### 4.4.1 SR (Set-priority Bistable Circuit)

This is a function block that realizes a bistable (flip-flop) circuit. The priority is given to the set input.

#### ■ Icon



#### ■ Parameter

| Scope  | Name  | Type | Description   |
|--------|-------|------|---|
| Input  | SET1  | BOOL | Specifies the set input for a bistable circuit.   |
|        | RESET | BOOL | Specifies the reset input for a bistable circuit.   |
| Output | Q1    | BOOL | When the input argument SET1 becomes TRUE, outputs and holds TRUE.<br>When the input argument RESET becomes TRUE, outputs and holds FALSE.<br>When both SET1 and RESET1 are TRUE, outputs and holds TRUE. |

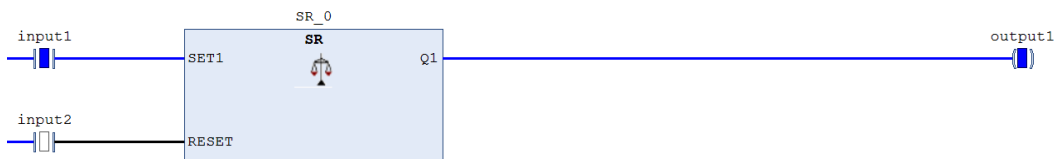
#### ■ Program example

When the input variable “input1” becomes TRUE, the output variable “output1” becomes TRUE. Even if the input variable “input1” becomes FALSE, “output1” remains TRUE.

When the input variable “input1” is FALSE and if input variable “input2” becomes TRUE, the output variable “output1” becomes FALSE.

The instance name is SR\_0.

#### LD program

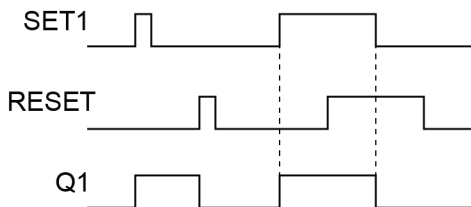


#### ST program

```
SR_0 (
  SET1 TRUE := input1 TRUE ,
  RESET FALSE := input2 FALSE ,
  Q1 TRUE => output1 TRUE ) ;
```

## 4.4 Bistable Circuit Instructions

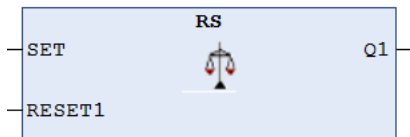
### ■ Time-sequence diagram



### 4.4.2 RS (Reset-priority Bistable Circuit)

This is a function block that realizes a bistable (flip-flop) circuit. The priority is given to the reset input.

### ■ Icon



### ■ Parameter

| Scope  | Name  | Type | Description  |
|--------|-------|------|--|
| Input  | SET1  | BOOL | Specifies the set input for a bistable circuit.  |
|        | RESET | BOOL | Specifies the reset input for a bistable circuit.  |
| Output | Q1    | BOOL | When the input argument SET1 becomes TRUE, outputs and holds TRUE.<br>When the input argument RESET becomes TRUE, outputs and holds FALSE.<br>When both SET1 and RESET1 are TRUE, outputs and holds FALSE. |

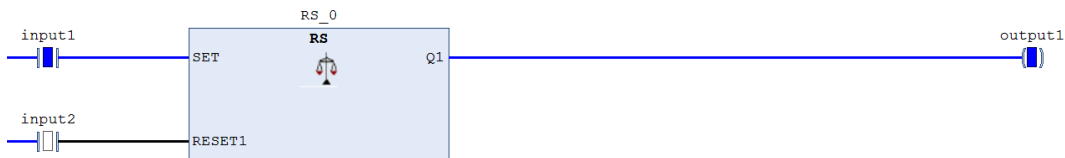
### ■ Program example

When the input variable “input1” becomes TRUE, the output variable “output1” becomes TRUE. Even if the input variable “input1” becomes FALSE, “output1” remains TRUE.

When the input variable “input1” is FALSE and if the input variable “input2” becomes TRUE, the output variable “output1” becomes FALSE.

The instance name is RS\_0.

### LD program





(MEMO)

# 5 Motion Control Function Blocks (Single Axis Control)

---

This section describes motion control function blocks for the single axis.

|   |      |
|---|------|
| 5.1 Servo ON .....  | 5-2  |
| 5.1.1 MC_Power (Servo ON) .....   | 5-2  |
| 5.2 Home Return.....  | 5-4  |
| 5.2.1 PMC_Home (Home Return).....   | 5-4  |
| 5.2.2 MC_Home (Home Return) .....   | 5-6  |
| 5.3 Control Switch.....   | 5-8  |
| 5.3.1 SMC_SetControllerMode (Control Mode Setting).....                       | 5-8  |
| 5.4 Stop.....   | 5-9  |
| 5.4.1 MC_Stop (Forced Stop) .....   | 5-9  |
| 5.4.2 MC_Halt (Halt) .....  | 5-10 |
| 5.5 JOG / Inching .....   | 5-11 |
| 5.5.1 MC_Jog (Jogging).....   | 5-11 |
| 5.5.2 SMC_Inch (Inching) .....  | 5-12 |
| 5.6 Position Control.....   | 5-14 |
| 5.6.1 MC_MoveAbsolute (Absolute Value Positioning).....                       | 5-14 |
| 5.6.2 MC_MoveRelative (Relative Value Positioning).....                       | 5-15 |
| 5.6.3 MC_MoveAdditive (Target Position Change) .....                          | 5-16 |
| 5.6.4 MC_MoveSuperImposed (Superimposed positioning) .....                    | 5-18 |
| 5.6.5 MC_PositionProfile (Position Profile Move) .....                        | 5-20 |
| 5.6.6 Default Setting for Variables of the MC_TP_REF Type Structure....       | 5-22 |
| 5.6.7 SMC_MoveContinuousAbsolute (Absolute Value Position Velocity Move)..... | 5-24 |
| 5.6.8 SMC_MoveContinuousRelative (Relative Value Position Velocity Move)..... | 5-25 |
| 5.7 Velocity Control .....  | 5-27 |
| 5.7.1 MC_MoveVelocity (Velocity Control).....                                 | 5-27 |
| 5.7.2 MC_VelocityProfile (Velocity Profile Movement) .....                    | 5-28 |
| 5.7.3 MC_AccelerationProfile (Acceleration Profile Movement) .....            | 5-29 |
| 5.8 Torque Control .....  | 5-32 |
| 5.8.1 PMC_SetTorque (Torque Control).....                                     | 5-32 |
| 5.8.2 SMC_SetTorque (Torque Control).....                                     | 5-33 |

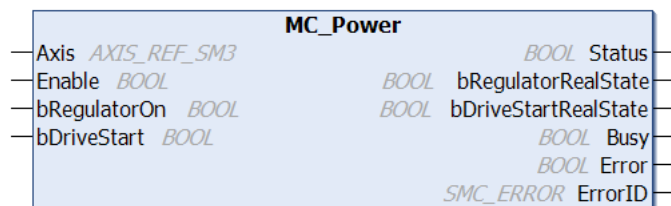
## 5.1 Servo ON

### 5.1 Servo ON

#### 5.1.1 MC\_Power (Servo ON)

This is a function block (FB) that sets the axis to the servo ON state to be ready for operation.

##### ■ Icon

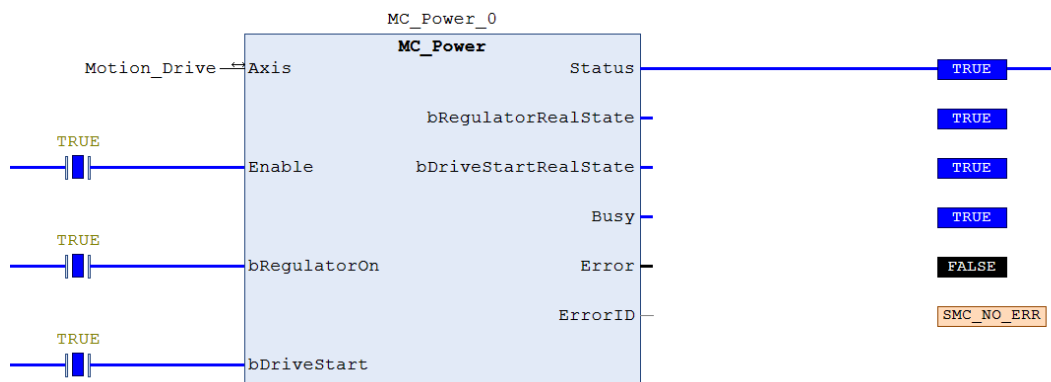


##### ■ Parameter

| Scope          | Name                 | Type         | Initial | Description   |
|----------------|----------------------|--------------|---------|---|
| Input / output | Axis                 | AXIS_REF_SM3 | -       | Specifies the axis.   |
| Input          | Enable               | BOOL         | FALSE   | TRUE: The FB can be executed.   |
|                | bRegulatorOn         | BOOL         | FALSE   | TRUE: Servo ON<br>FALSE: Servo OFF                                    |
|                | bDriveStart          | BOOL         | FALSE   | TRUE: Quick stop is disabled.<br>For the GM1 Controller, fix to TRUE. |
| Output         | Status               | BOOL         | FALSE   | TRUE: The axis can be executed.                                       |
|                | bRegulatorRealState  | BOOL         | FALSE   | TRUE: The FB is ready to be executed.                                 |
|                | bDriveStartRealState | BOOL         | FALSE   | TRUE: Operation is not stopped due to quick stop.                     |
|                | Busy                 | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                           |
|                | Error                | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                            |
|                | ErrorID              | SMC_ERROR    | FALSE   | An error ID is output.  |

##### ■ Program example

The following program is designed to set the axis that corresponds to the input variable Motion\_Drive to the servo ON state.

**i Info.**

- If the input arguments Enable, bRegulatorOn, and bDriveStart are TRUE and the output argument Status is FALSE, a hardware problem may occur.
- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*

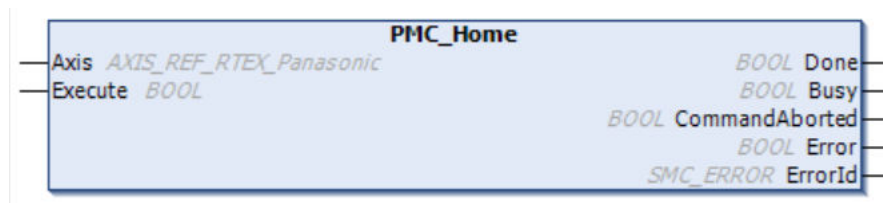
## 5.2 Home Return

### 5.2 Home Return

#### 5.2.1 PMC\_Home (Home Return)

This is a function block (FB) that performs home return of the axis. The home return function of the servo amplifier is used.

##### ■ Icon



##### ■ Parameter

| Scope          | Name           | Type                        | Initial | Description  |
|----------------|----------------|-----------------------------|---------|--|
| Input / output | Axis           | AXIS_REF_RTE<br>X_Panasonic | -       | Specifies the axis.  |
| Input          | Execute        | BOOL                        | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
| Output         | Done           | BOOL                        | FALSE   | TRUE: Execution is completed and transitioned to the Standstill state. |
|                | Busy           | BOOL                        | FALSE   | TRUE: Execution of the FB is not completed.                            |
|                | CommandAborted | BOOL                        | FALSE   | TRUE: An interruption from other FB has occurred.                      |
|                | Error          | BOOL                        | FALSE   | TRUE: An error has occurred within the FB.                             |
|                | ErrorID        | SMC_ERROR                   | 0       | An error ID is output.   |

##### ■ Execution operation

- Execute = TRUE: Starts the home return mode. Execute = FALSE: Ends the home return mode.
- When PMC\_Home is successfully completed (when Done changes to TRUE), the home return mode is automatically ended.
- When PMC\_Home is abnormally terminated (when Error changes to TRUE), end the home return mode by setting Execute = FALSE and making a call.

##### ■ Execution errors

The PMC\_Home function block outputs the following errors.



| Error                                 | Description   |
|---------------------------------------|---|
| SMC_WRONG_CONTROLLER_MODE             | Executed in a mode other than the position control mode.<br>Change to SMC_position using SMC_SetControllerMode. |
| SMC_DI_HOMING_ERROR                   | The version of the amplifier paired with an absolute encoder is lower than V1.24.                               |
|                                       | Trigger setting is incorrect.   |
|                                       | Amplifier parameters (Pr4.00 to Pr4.07) are incorrect.  |
|                                       | Abnormal state in HOME, POT, or NOT is detected.  |
|                                       | The home return cannot be completed even if POT and NOT settings were inverted three times or more.             |
|                                       | The home return was completed at an incorrect position.   |
| SMC_MS_DIRECTION_NOT_APPLICABLE       | The return direction setting is incorrect.  |
| SMC_AXIS_NOT_READY_FOR_MOTION         | The axis is in a state (Stopping, Disabled, or Errorstop) where PMC_Home cannot be executed.                    |
| SMC_REGULATOR_OR_START_NOT_SET        | The servo turns OFF and the brake was applied.  |
| SMC_3SH_INVALID_VELACC_VALUES         | The input target velocity, home return creep speed, acceleration, or deceleration is incorrect.                 |
| SMC_AXIS_REF_CHANGED_DURING_OPERATION | The Axis was changed during operation.  |

#### ■ Execution conditions

- As the PMC\_Home function block uses the RTEX home return command, it cannot be executed together with PMC\_ReadLatchPosition or PMC\_StopLatchPosition.
- If PMC\_Home is executed while PMC\_ReadLatchPosition or PMC\_StopLatchPosition is being executed, the CommandAborted parameter becomes TRUE. Furthermore, if PMC\_Home of another instance is executed while one PMC\_Home is being executed, the CommandAborted parameter of the PMC\_Home executed later becomes TRUE.

#### ■ Amplifier parameter conditions

When using PMC\_Home, set amplifier parameters as shown in the following table.

| Parameter | Parameter name      | Setting A | Setting B |
|-----------|---------------------|-----------|-----------|
| Pr4.00    | SI1 input selection | SI-MON5   | SI-MON5   |
| Pr4.01    | SI2 input selection | POT       |           |
| Pr4.02    | SI3 input selection | NOT       |           |
| Pr4.03    | SI4 input selection | SI-MON1   | SI-MON1   |
| Pr4.04    | SI5 input selection | HOME      | HOME      |
| Pr4.05    | SI6 input selection | EXT2      | POT       |
| Pr4.06    | SI7 input selection | EXT3      | NOT       |
| Pr4.07    | SI8 input selection | SI-MON4   | SI-MON4   |

Return methods that can be executed for the settings A and B are as shown in the following table.

## 5.2 Home Return

| Return method                 | Setting A                                 | Setting B                      |
|-------------------------------|---|--------------------------------|
| DOG method 1                  | <input type="radio"/>                     | <input type="radio"/> (Note 2) |
| DOG method 2                  | <input checked="" type="radio"/> (Note 1) | <input type="radio"/>          |
| DOG method 3                  | <input type="radio"/>                     | <input type="radio"/> (Note 2) |
| Limit method 1                | <input type="radio"/>                     | <input type="radio"/> (Note 2) |
| Limit method 2                | <input checked="" type="radio"/> (Note 1) | <input type="radio"/>          |
| Home return method            | <input type="radio"/>                     | <input type="radio"/> (Note 2) |
| Stop-on-contact method 1      | <input type="radio"/>                     | <input type="radio"/> (Note 2) |
| Stop-on-contact method 2      | <input type="radio"/>                     | <input type="radio"/>          |
| Data setting method           | <input type="radio"/>                     | <input type="radio"/>          |
| High-speed home return method | <input type="radio"/>                     | <input type="radio"/>          |

(Note 1) When using POT, NOT, or HOME as a home reference trigger, assign them as follows.

HOME: SI5 input selection

POT: SI6 input selection

NOT: SI7 input selection

(Note 2) When EXT2 or EXT3 is used as a home reference trigger, it can be used only for the above setting A.

### **i** Info.

- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*

### 5.2.2 MC\_Home (Home Return)

This is a function block (FB) that performs home return.

#### ■ Icon



#### ■ Parameter

| Scope          | Name    | Type         | Default | Description                                |
|----------------|---------|--------------|---------|--|
| Input / output | Axis    | AXIS_REF_SM3 | -       | Specifies the axis.                        |
| Input          | Execute | BOOL         | FALSE   | TRUE: Starts execution at the rising edge. |

---

| Scope  | Name           | Type      | Default | Description  |
|--------|----------------|-----------|---------|--|
|        | Position       | LREAL     | 0       | Set value of the absolute position when the reference signal is detected |
| Output | Done           | BOOL      | FALSE   | TRUE: Stopping is completed.   |
|        | Busy           | BOOL      | FALSE   | TRUE: Execution of the FB is not completed.                              |
|        | CommandAborted | BOOL      | FALSE   | TRUE: An interruption from other FB has occurred.                        |
|        | Error          | BOOL      | FALSE   | TRUE: An error has occurred within the FB.                               |
|        | ErrorID        | SMC_ERROR | 0       | Error ID output  |

## 5.3 Control Switch

### 5.3 Control Switch

#### 5.3.1 SMC\_SetControllerMode (Control Mode Setting)

This is a function block (FB) that sets up the control mode for controlling the position, velocity, and torque.

##### ■ Icon



##### ■ Parameter

| Scope          | Name            | Type                | Initial      | Description                                  |
|----------------|-----------------|---------------------|--------------|--|
| Input / output | Axis            | AXIS_REF_SM3        | -            | Specifies the axis.                          |
| Input          | bExecute        | BOOL                | FALSE        | Sets up the control mode at the rising edge. |
|                | nControllerMode | SMC_CONTROLLER_MODE | SMC_position | Specifies the control mode.                  |
| Output         | bDone           | BOOL                | FALSE        | TRUE: Control mode setup is completed.       |
|                | bBusy           | BOOL                | FALSE        | TRUE: Execution of the FB is not completed.  |
|                | bError          | BOOL                | FALSE        | TRUE: An error has occurred within the FB.   |
|                | nErrorID        | SMC_ERROR           | 0            | An error ID is output.                       |

##### ■ SMC\_CONTROLLER\_MODE (Enumeration type)

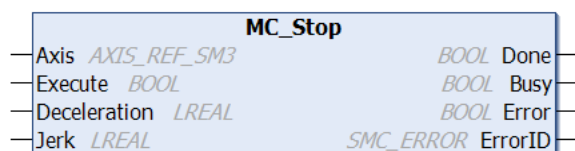
| Name          | Value | Description      |
|---------------|-------|------------------|
| SMC_nocontrol | 0     | Usage prohibited |
| SMC_torque    | 1     | Torque mode      |
| SMC_velocity  | 2     | Velocity mode    |
| SMC_position  | 3     | Position mode    |
| SMC_current   | 4     | Usage prohibited |

## 5.4 Stop

### 5.4.1 MC\_Stop (Forced Stop)

This is a function block (FB) that causes the axis to make a deceleration stop. After stopping, the axis remains stopped while Execute is TRUE. While the axis is being decelerated or while it is stopped, other function blocks cannot be executed.

#### ■ Icon



#### ■ Parameter

| Scope          | Name         | Type         | Initial | Description   |
|----------------|--------------|--------------|---------|---|
| Input / output | Axis         | AXIS_REF_SM3 | -       | Specifies the axis.   |
| Input          | Execute      | BOOL         | FALSE   | Starts execution at the rising edge. While it is TRUE, other FB cannot be executed. |
|                | Deceleration | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).                                     |
|                | Jerk         | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).   |
| Output         | Done         | BOOL         | FALSE   | TRUE: The axis velocity has reached 0.  |
|                | Busy         | BOOL         | FALSE   | TRUE: The FB is in operation.   |
|                | Error        | BOOL         | FALSE   | TRUE: An error has occurred.  |
|                | ErrorID      | SMC_ERROR    | 0       | An error ID is output.  |

#### Info.

- In the torque control mode (SMC\_torque), the axis cannot be stopped using MC\_Stop. For stopping methods, refer to PMC\_SetTorque.
- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

#### REFERENCE

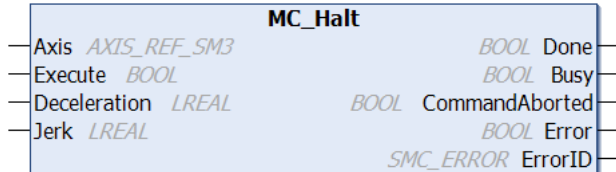
##### 5.8.1 PMC\_SetTorque (Torque Control)

## 5.4 Stop

### 5.4.2 MC\_Halt (Halt)

This is a function block (FB) that causes the axis to make a deceleration stop. After the axis is stopped or while the axis is being decelerated, other motion instructions can be executed.

#### ■ Icon



#### ■ Parameter

| Scope          | Name           | Type         | Initial | Description                                       |
|----------------|----------------|--------------|---------|---|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.                               |
| Input          | Execute        | BOOL         | FALSE   | Starts execution at the rising edge.              |
|                | Deceleration   | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).   |
|                | Jerk           | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).           |
| Output         | Done           | BOOL         | FALSE   | TRUE: The axis velocity has reached 0.            |
|                | Busy           | BOOL         | FALSE   | TRUE: The FB is in operation.                     |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred. |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred.                      |
|                | ErrorID        | SMC_ERROR    | 0       | Error ID output                                   |

#### **i** Info.

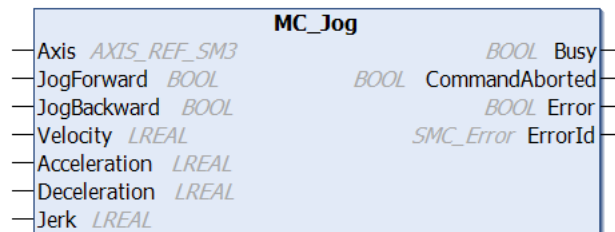
- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

## 5.5 JOG / Inching

### 5.5.1 MC\_Jog (Jogging)

This is a function block (FB) that causes the axis to keep traveling in a forward or reverse direction at a constant velocity. While the input is TRUE, the axis keeps traveling in a forward or reverse direction at a constant velocity.

#### ■ Icon



#### ■ Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | JogForward     | BOOL         | FALSE   | While JogForward is TRUE, the axis travels in a forward direction. If JogBackward is TRUE at the same time, the axis does not operate. |
|                | JogBackward    | BOOL         | FALSE   | While JogBackward is TRUE, the axis travels in a reverse direction. If JogForward is TRUE at the same time, the axis does not operate. |
|                | Velocity       | LREAL        | 0       | Specifies the velocity (u/s).  |
|                | Acceleration   | LREAL        | 0       | Specifies the acceleration (u/s <sup>2</sup> ).  |
|                | Deceleration   | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).  |
|                | Jerk           | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).  |
| Output         | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.  |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.  |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.   |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.   |

## 5.5 JOG / Inching

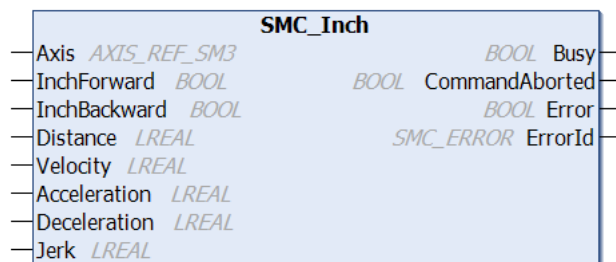
### **i** Info.

- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

### 5.5.2 SMC\_Inch (Inching)

This is a function block (FB) that causes the axis to travel in a forward or reverse direction for a specified relative distance. When the input turns TRUE, the axis travels in a forward or reverse direction for a specified relative distance.

#### ■ Icon



#### ■ Parameter

| Scope          | Name         | Type         | Initial | Description  |
|----------------|--------------|--------------|---------|--|
| Input / output | Axis         | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | InchForward  | BOOL         | FALSE   | When the input changes from FALSE to TRUE, the axis starts traveling in a forward direction for the distance specified in Distance. When the input changes to FALSE before the axis travels the specified distance, the axis stops traveling. If InchBackward is TRUE at the same time, the axis does not operate. |
|                | InchBackward | BOOL         | FALSE   | When the input changes from FALSE to TRUE, the axis starts traveling in a reverse direction for the distance specified in Distance. When the input changes to FALSE before the axis travels the specified distance, the axis stops traveling. If InchForward is TRUE at the same time, the axis does not operate.  |
|                | Distance     | LREAL        | 0       | Specifies the travel distance (u).   |
|                | Velocity     | LREAL        | 0       | Specifies the velocity (u/s).  |
|                | Acceleration | LREAL        | 0       | Specifies the acceleration (u/s <sup>2</sup> ).  |
|                |              | Deceleration | LREAL   | 0  |



| Scope  | Name           | Type      | Initial | Description                                       |
|--------|----------------|-----------|---------|---|
|        | Deceleration   | LREAL     | 0       | Specifies the deceleration ( $u/s^2$ ).           |
|        | Jerk           | LREAL     | 0       | Specifies the jerk ( $u/s^3$ ).                   |
| Output | Busy           | BOOL      | FALSE   | TRUE: Execution of the FB is not completed.       |
|        | CommandAborted | BOOL      | FALSE   | TRUE: An interruption from other FB has occurred. |
|        | Error          | BOOL      | FALSE   | TRUE: An error has occurred within the FB.        |
|        | ErrorID        | SMC_ERROR | 0       | An error ID is output.                            |

### Info.

- Reference manual
  - GM1 Controller RTEX User's Manual (Operation Edition)*
  - GM1 Controller EtherCAT User's Manual (Operation Edition)*

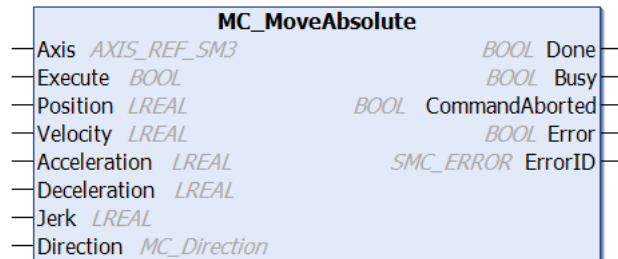
## 5.6 Position Control

### 5.6 Position Control

#### 5.6.1 MC\_MoveAbsolute (Absolute Value Positioning)

This is a function block (FB) that causes the axis to travel to a position specified as an absolute position.

##### ■ Icon



##### ■ Parameter

| Scope          | Name           | Type         | Initial  | Description   |
|----------------|----------------|--------------|----------|---|
| Input / output | Axis           | AXIS_REF_SM3 | -        | Specifies the axis.   |
| Input          | Execute        | BOOL         | FALSE    | Starts execution at the rising edge.  |
|                | Position       | LREAL        | 0        | Specifies the target position (u).  |
|                | Velocity       | LREAL        | 0        | Specifies the maximum velocity (u/s).   |
|                | Acceleration   | LREAL        | 0        | Specifies the acceleration (u/s <sup>2</sup> ).   |
|                | Deceleration   | LREAL        | 0        | Specifies the deceleration (u/s <sup>2</sup> ).   |
|                | Jerk           | LREAL        | 0        | Specifies the jerk (u/s <sup>3</sup> ).   |
|                | Direction      | MC_Direction | shortest | Specifies the traveling direction of the axis.<br>Direction can be specified only for the modulo type. For the finite axis, the specification is ignored. |
| Output         | Done           | BOOL         | FALSE    | TRUE: The target position has been reached.   |
|                | Busy           | BOOL         | FALSE    | TRUE: The FB is in operation.   |
|                | CommandAborted | BOOL         | FALSE    | TRUE: An interruption from other FB has occurred.   |
|                | Error          | BOOL         | FALSE    | TRUE: An error has occurred.  |
|                | ErrorID        | SMC_ERROR    | 0        | An error ID is output.  |

### ■ MC\_Direction (Enumeration type)

| Name     | Value | Description   |
|----------|-------|---|
| positive | 1     | Travels in the positive direction.  |
| negative | -1    | Travels in the negative direction.  |
| shortest | 0     | Travels in the shortest direction from the current command position to the target command position when MC_MoveAbsolute is executed.  |
| fastest  | 3     | Travels in the fastest direction from the current command position to the target command position when MC_MoveAbsolute is executed.<br>If the axis is being driven by another function block when MC_MoveAbsolute is executed, selects the fastest direction within the GM1.<br>If the axis is being stopped when MC_MoveAbsolute is executed, makes the same movement as for the shortest. |
| current  | 2     | Travels to the current direction.<br>If the axis is being driven by another function block when MC_MoveAbsolute is executed, travels in the same direction.<br>If the axis is being stopped when MC_MoveAbsolute is executed, travels in the direction moved by the previously executed function block.   |

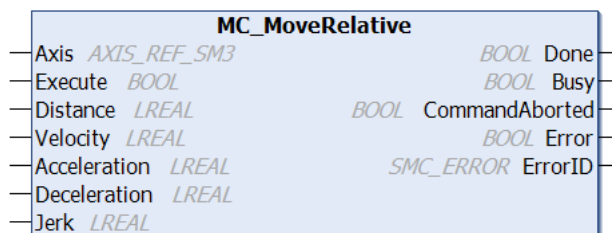
#### Info.

- Reference manual
  - GM1 Controller RTEX User's Manual (Operation Edition)*
  - GM1 Controller EtherCAT User's Manual (Operation Edition)*

### 5.6.2 MC\_MoveRelative (Relative Value Positioning)

This is a function block (FB) that causes the axis to travel to a position specified as a relative position.

#### ■ Icon



## 5.6 Position Control

### ■ Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute        | BOOL         | FALSE   | Starts execution at the rising edge.                         |
|                | Distance       | LREAL        | 0       | Specifies the relative distance (u).                         |
|                | Velocity       | LREAL        | 0       | Specifies the maximum velocity (u/s).                        |
|                | Acceleration   | LREAL        | 0       | Specifies the acceleration (u/s <sup>2</sup> ).              |
|                | Deceleration   | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).              |
|                | Jerk           | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).                      |
| Output         | Done           | BOOL         | FALSE   | TRUE: The axis has traveled the specified relative distance. |
|                | Busy           | BOOL         | FALSE   | TRUE: FB operation is in progress.                           |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.            |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred.                                 |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.                                       |

### Info.

- Reference manual
  - GM1 Controller RTEX User's Manual (Operation Edition)*
  - GM1 Controller EtherCAT User's Manual (Operation Edition)*

### 5.6.3 MC\_MoveAdditive (Target Position Change)

This is a function block (FB) that adds a relative distance to the target position of the immediately preceding instruction. No addition is made to the velocity, acceleration, or deceleration; they change to the specified values.

### ■ Icon



### ■ Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute        | BOOL         | FALSE   | Starts execution at the rising edge.                         |
|                | Distance       | LREAL        | 0       | Specifies the relative distance (u) to be added.             |
|                | Velocity       | LREAL        | 0       | Specifies the maximum velocity (u/s).                        |
|                | Acceleration   | LREAL        | 0       | Specifies the acceleration (u/s <sup>2</sup> ).              |
|                | Deceleration   | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).              |
|                | Jerk           | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).                      |
| Output         | Done           | BOOL         | FALSE   | TRUE: The axis has traveled the specified relative distance. |
|                | Busy           | BOOL         | FALSE   | TRUE: The FB is in operation.                                |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.            |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred.                                 |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.                                       |

### ■ Operations when the function block is executed

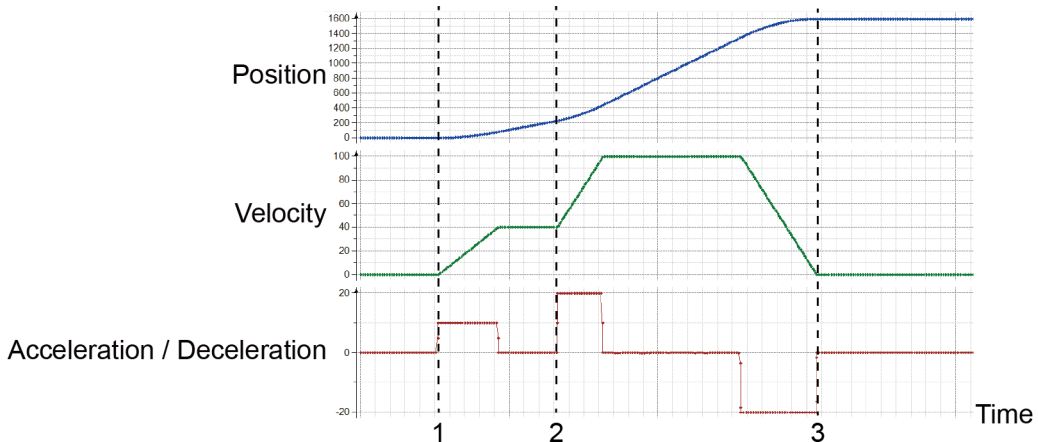
This example shows the position, velocity, acceleration, and deceleration traces when MC\_MoveAdditive is executed while MC\_MoveRelative (Relative Value Positioning) is being executed.

#### Function block input parameters

| Execution sequence | Function blocks | Distance (Relative distance) | Velocity (Velocity) | Acceleration (Acceleration) | Deceleration (Deceleration) |
|--------------------|-----------------|------------------------------|---------------------|-----------------------------|-----------------------------|
| 1                  | MC_MoveRelative | 1000                         | 40                  | 10                          | 10                          |
| 2                  | MC_MoveAdditive | 600                          | 100                 | 20                          | 20                          |

## 5.6 Position Control

### Trace



1. MC\_MoveRelative is started.
2. MC\_MoveAdditive is started.

MC\_MoveRelative is interrupted at the timing when MC\_MoveAdditive is started. The output parameter CommandAborted of MC\_MoveRelative turns TRUE.

The velocity and acceleration / deceleration change to values (velocity: 100, acceleration / deceleration: 20) specified by MC\_MoveAdditive.

3. MC\_MoveAdditive is completed.

The axis travels to the position where the relative distance specified by MC\_MoveAdditive is added to the position specified by MC\_MoveRelative (1000+600=1600).

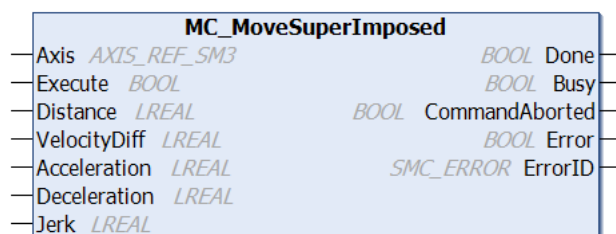
### **i** Info.

- Use the MC\_MoveSuperImposed function block to cause the axis to travel after adding the velocity, acceleration, and deceleration to the previously executed instruction.
- By setting Distance = 0, it is possible to change the velocity, acceleration, deceleration, and jerk during axis operation.

### 5.6.4 MC\_MoveSuperImposed (Superimposed positioning)

This is a function block (FB) that adds a relative distance, a velocity, an acceleration, and a deceleration to the operations of the immediately preceding instruction.

#### ■ Icon



## ■ Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute        | BOOL         | FALSE   | Starts execution at the rising edge.                         |
|                | Distance       | LREAL        | 0       | Specifies the relative distance (u).                         |
|                | VelocityDiff   | LREAL        | 0       | Specifies the maximum velocity (u/s).                        |
|                | Acceleration   | LREAL        | 0       | Specifies the acceleration (u/s <sup>2</sup> ).              |
|                | Deceleration   | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).              |
|                | Jerk           | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).                      |
| Output         | Done           | BOOL         | FALSE   | TRUE: The axis has traveled the specified relative distance. |
|                | Busy           | BOOL         | FALSE   | TRUE: The FB is in operation.                                |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.            |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred.                                 |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.                                       |

### Note

- When MC\_MoveSuperImposed is being executed (Busy = TRUE), be sure to call at every interval.

If a call is not made, the axis may perform an unexpected operation.

## ■ Operations when the function block is executed

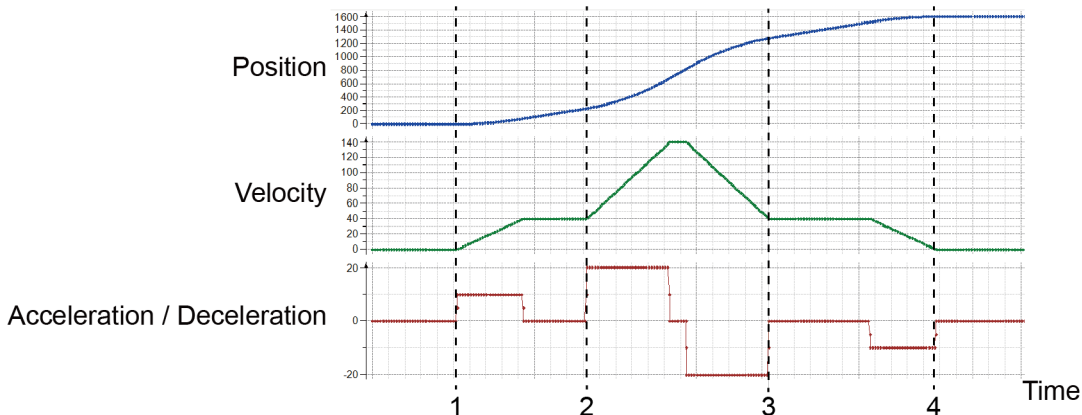
This example shows the position, velocity, and acceleration traces when MC\_MoveSuperImposed is executed while MC\_MoveRelative (Relative Value Positioning) is being executed.

### Function block input parameters

| Execution sequence | Function blocks     | Distance (Relative distance) | Velocity (Velocity) | Acceleration (Acceleration) | Deceleration (Acceleration) |
|--------------------|---------------------|------------------------------|---------------------|-----------------------------|-----------------------------|
| 1                  | MC_MoveRelative     | 1000                         | 40                  | 10                          | 10                          |
| 2                  | MC_MoveSuperImposed | 600                          | 100                 | 20                          | 20                          |

## 5.6 Position Control

### Trace



1. MC\_MoveRelative is started.
2. MC\_MoveSuperImposed is started.  
The specified velocity, acceleration, and deceleration are added at the timing when MC\_MoveSuperImposed is started.
3. MC\_MoveSuperImposed is completed.
4. MC\_MoveRelative is completed.  
The axis travels to the position where the relative distance specified by MC\_MoveSuperImposed is added to the position specified by MC\_MoveRelative (1000+600=1600).

### **i** Info.

- Use the MC\_MoveAdditive function block to cause the axis to travel at the specified values without adding the velocity, acceleration, and deceleration to the previously executed instruction.

### 5.6.5 MC\_PositionProfile (Position Profile Move)

This is a function block (FB) that causes the axis to operate according to the profile data that consists of a combination of position and time.

#### ■ Icon





### ■ Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
|                | TimePosition   | MC_TP_REF    | -       | Specifies the time/position profile.   |
| Input          | Execute        | BOOL         | FALSE   | Starts execution at the rising edge.   |
|                | ArraySize      | INT          | 0       | A member of the Input / output TimePosition.<br>Specifies the number of points to be executed by FB in the array of time and position that are specified by MC_TP_ARRAY. |
|                | PositionScale  | LREAL        | 1       | Position scaling   |
|                | Offset         | LREAL        | 0       | Position offset  |
| Output         | Done           | BOOL         | FALSE   | TRUE: The movements specified by the profile are completed.  |
|                | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.  |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.  |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.   |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.   |

### ■ MC\_TP\_REF (Structure)

| Member          | Type                     | Description  |
|-----------------|--------------------------|--|
| Number_of_pairs | INT                      | Not used   |
| isAbsolute      | BOOL                     | Methods of specifying the position of profile data<br>TRUE: Specified in an absolute value.<br>FALSE: Specified in a relative value. |
| MC_TP_Array     | ARRAY [1..100] OF SMC_TP | Time and position profile data<br>(1st point to 100th point)   |

### ■ SMC\_TP (Structure)

| Member     | Type  | Description                  |
|------------|-------|------------------------------|
| delta_time | TIME  | Time of the profile data     |
| position   | LREAL | Position of the profile data |

### ■ Operations when the function block is executed

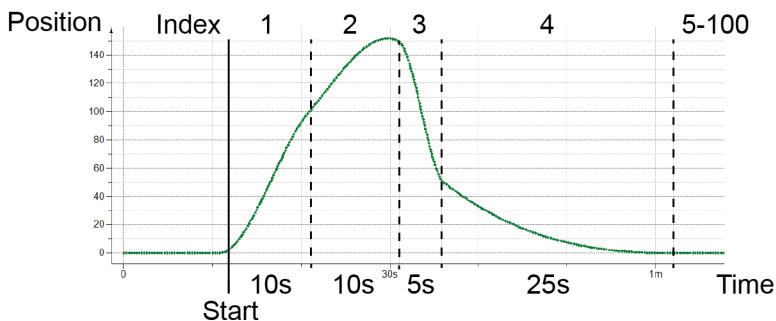
This example shows time and position traces when the MC\_PositionProfile function block is executed with the following parameter settings.

## 5.6 Position Control

### Parameter

| Index of the MC_TP_Array array, a member of the input TimePosition | Delta_time (time) | Position (absolute position) |
|--|-------------------|------------------------------|
| 1  | Time#10s          | 100                          |
| 2  | Time#10s          | 150                          |
| 3  | Time#5s           | 50                           |
| 4  | Time#25s          | 0                            |
| 5 to 100   | Time#0ms          | 0                            |

### Trace



### Note

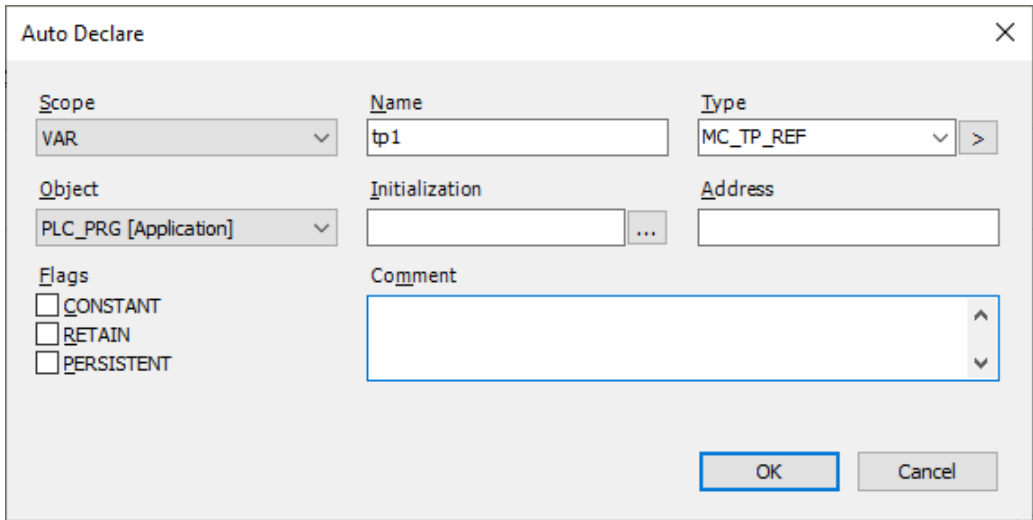
- While the axis keeps driving, do not specify delta\_time to 0 ms. Otherwise, it may cause an abnormal operation.

### 5.6.6 Default Setting for Variables of the MC\_TP\_REF Type Structure

To enter the value of the input TimePosition, it is necessary to make default setting for variables of the MC\_TP\_REF type structure.

#### 1.2 Procedure

1. When the input variable to TimePosition is declared, "Automatic Declaration" dialog box is displayed. Click  displayed next to the "Initial Value" field.

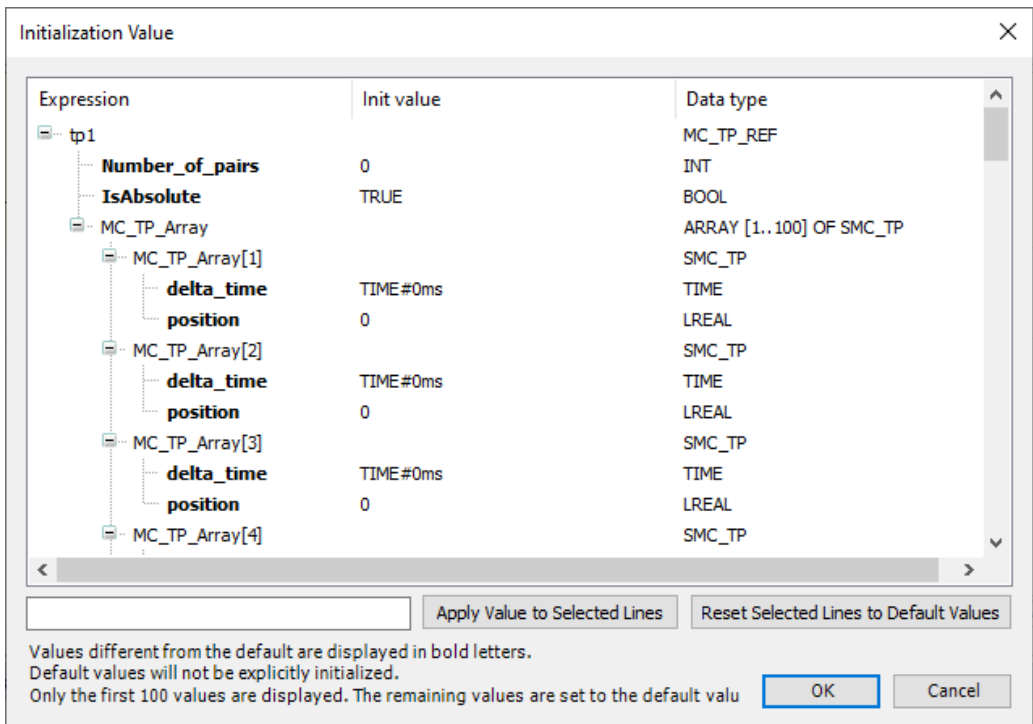


The "Auto Declare" dialog box is used to define a variable. It contains the following fields:

- Scope:** VAR
- Name:** tp1
- Type:** MC\_TP\_REF
- Object:** PLC\_PRG [Application]
- Initialization:** (empty field with a browse button "...")
- Address:** (empty field)
- Flags:**
  - CONSTANT
  - RETAIN
  - PERSISTENT
- Comment:** (empty text area)

Buttons: OK, Cancel

- The "Initial Value" dialog box is displayed and, on the dialog box, you can set the default value for every member of the variable type (MC\_TP\_REF).



The "Initialization Value" dialog box displays a tree view of the variable structure and a table of initialization values.

| Expression      | Init value | Data type                |
|-----------------|------------|--------------------------|
| tp1             |            | MC_TP_REF                |
| Number_of_pairs | 0          | INT                      |
| IsAbsolute      | TRUE       | BOOL                     |
| MC_TP_Array     |            | ARRAY [1..100] OF SMC_TP |
| MC_TP_Array[1]  |            | SMC_TP                   |
| delta_time      | TIME#0ms   | TIME                     |
| position        | 0          | LREAL                    |
| MC_TP_Array[2]  |            | SMC_TP                   |
| delta_time      | TIME#0ms   | TIME                     |
| position        | 0          | LREAL                    |
| MC_TP_Array[3]  |            | SMC_TP                   |
| delta_time      | TIME#0ms   | TIME                     |
| position        | 0          | LREAL                    |
| MC_TP_Array[4]  |            | SMC_TP                   |

Buttons: Apply Value to Selected Lines, Reset Selected Lines to Default Values

Values different from the default are displayed in bold letters.  
 Default values will not be explicitly initialized.  
 Only the first 100 values are displayed. The remaining values are set to the default value

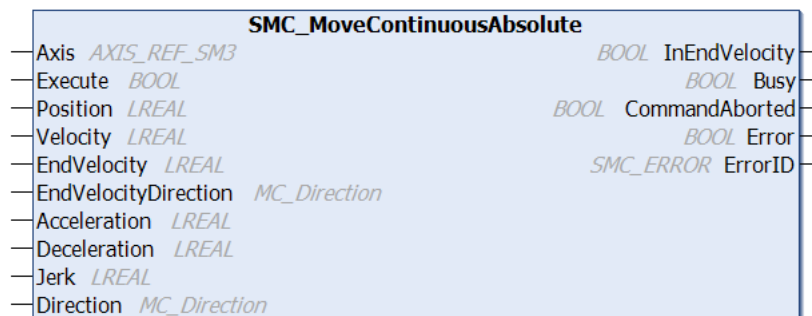
Buttons: OK, Cancel

## 5.6 Position Control

### 5.6.7 SMC\_MoveContinuousAbsolute (Absolute Value Position Velocity Move)

This is a function block (FB) that executes absolute value positioning and, after the axis reaches the target position, causes the axis to keep moving at a specified velocity.

#### ■ Icon



#### ■ Parameter

| Scope          | Name                 | Type         | Initial  | Description  |
|----------------|----------------------|--------------|----------|--|
| Input / output | Axis                 | AXIS_REF_SM3 | -        | Specifies the axis.  |
| Input          | Execute              | BOOL         | FALSE    | Starts execution at the rising edge.   |
|                | Position             | LREAL        | 0        | Specifies the target position (u).   |
|                | Velocity             | LREAL        | 0        | Specifies the velocity (u/s) until the axis reaches the target position.   |
|                | EndVelocity          | LREAL        | 0        | Specifies the velocity (u/s) after the axis reaches the target position.   |
|                | EndVelocityDirection | MC_Direction | current  | Specifies the traveling direction after the axis reaches the target. Specifies either "positive", "negative", or "current". If "fastest" or "shortest" is specified, an error occurs.      |
|                | Acceleration         | LREAL        | 0        | Specifies the acceleration (u/s <sup>2</sup> ).  |
|                | Deceleration         | LREAL        | 0        | Specifies the deceleration (u/s <sup>2</sup> ).  |
|                | Jerk                 | LREAL        | 0        | Specifies the jerk (u/s <sup>3</sup> ).  |
| Output         | Direction            | MC_Direction | shortest | Specifies the traveling direction of the axis until the axis reaches the target position. Possible to specify only for the modulo type. For the finite axis, the specification is ignored. |
|                | InEndVelocity        | BOOL         | FALSE    | TRUE: The axis has reached the target position (Position).   |
|                | Busy                 | BOOL         | FALSE    | TRUE: Execution of the FB is not completed.  |

| Scope | Name           | Type      | Initial | Description                                       |
|-------|----------------|-----------|---------|---|
|       | CommandAborted | BOOL      | FALSE   | TRUE: An interruption from other FB has occurred. |
|       | Error          | BOOL      | FALSE   | TRUE: An error has occurred within the FB.        |
|       | ErrorID        | SMC_ERROR | 0       | An error ID is output.                            |

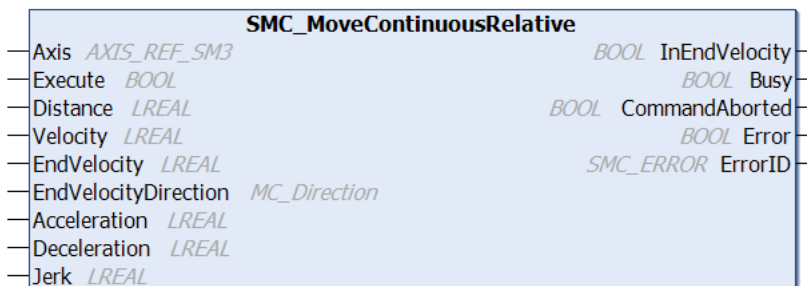
■ **MC\_Direction (Enumeration type)**

| Name     | Value | Description  |
|----------|-------|--|
| positive | 1     | Travels in the positive direction.   |
| negative | -1    | Travels in the negative direction.   |
| shortest | 0     | Travels in the shortest direction from the current command position to the target command position when SMC_MoveContinuousAbsolute is executed.  |
| fastest  | 3     | Travels in the fastest direction from the current command position to the target command position when SMC_MoveContinuousAbsolute is executed.<br>If the axis is being driven by another function block when SMC_MoveContinuousAbsolute is executed, selects the fastest direction within the GM1.<br>If the axis is being stopped when SMC_MoveContinuousAbsolute is executed, makes the same movement as for the shortest. |
| current  | 2     | Travels to the current direction.<br>If the axis is being driven by another function block when SMC_MoveContinuousAbsolute is executed, travels in the same direction.<br>If the axis is being stopped when SMC_MoveContinuousAbsolute is executed, travels in the direction moved by the previously executed function block.  |

**5.6.8 SMC\_MoveContinuousRelative (Relative Value Position Velocity Move)**

This is a function block (FB) that executes relative value positioning and, after the axis reaches the target position, causes the axis to keep moving at a specified velocity.

■ **Icon**



## 5.6 Position Control

### ■ Parameter

| Scope          | Name                 | Type         | Initial | Description  |
|----------------|----------------------|--------------|---------|--|
| Input / output | Axis                 | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute              | BOOL         | FALSE   | Starts execution at the rising edge.   |
|                | Distance             | LREAL        | 0       | Specifies the relative distance (u).   |
|                | Velocity             | LREAL        | 0       | Specifies the velocity (u/s).  |
|                | EndVelocity          | LREAL        | 0       | Specifies the velocity (u/s) after the axis travels the relative distance.   |
|                | EndVelocityDirection | MC_Direction | current | Specifies the traveling direction after the axis travels the relative distance.<br>Specifies either "positive", "negative", or "current".<br>If "fastest" or "shortest" is specified, an error occurs. |
|                | Acceleration         | LREAL        | 0       | Specifies the acceleration (u/s <sup>2</sup> ).  |
|                | Deceleration         | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).  |
|                | Jerk                 | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).  |
| Output         | InEndVelocity        | BOOL         | FALSE   | TRUE: The axis has traveled the specified relative distance and has reached the specified velocity.  |
|                | Busy                 | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.  |
|                | CommandAborted       | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.  |
|                | Error                | BOOL         | FALSE   | TRUE: An error has occurred within the FB.   |
|                | ErrorID              | SMC_ERROR    | 0       | An error ID is output.   |

### ■ MC\_Direction (Enumeration type)

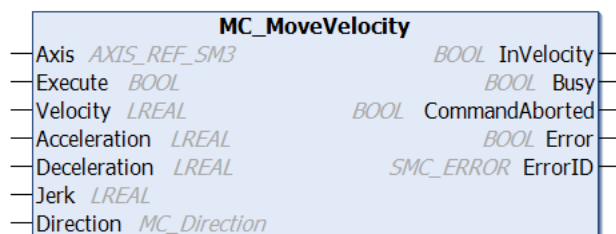
| Name     | Value | Description  |
|----------|-------|--|
| positive | 1     | Travels in the positive direction.   |
| negative | -1    | Travels in the negative direction.   |
| shortest | 0     | Not available. Do not specify this.  |
| fastest  | 3     | Not available. Do not specify this.  |
| current  | 2     | Travels to the current direction.<br>Possible to use only for the modulo axis. |

## 5.7 Velocity Control

### 5.7.1 MC\_MoveVelocity (Velocity Control)

This is a function block (FB) that specifies the velocity of the axis. The axis keeps moving at the specified velocity and direction.

#### ■ Icon



#### ■ Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute        | BOOL         | FALSE   | Starts execution at the rising edge.   |
|                | Velocity       | LREAL        | 0       | Specifies the velocity (u/s).  |
|                | Acceleration   | LREAL        | 0       | Specifies the acceleration (u/s <sup>2</sup> ).  |
|                | Deceleration   | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).  |
|                | Jerk           | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).  |
|                | Direction      | MC_Direction | current | Specifies the traveling direction of the axis.<br>"positive", "negative", or "current"<br>(An error occurs when "fastest" or "shortest" is selected) |
| Output         | InVelocity     | BOOL         | FALSE   | TRUE: The axis has reached the specified velocity for the first time.  |
|                | Busy           | BOOL         | FALSE   | TRUE: The FB is in operation.  |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.  |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred.   |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.   |

#### ■ MC\_Direction (Enumeration type)

| Name     | Value | Description                        |
|----------|-------|------------------------------------|
| positive | 1     | Travels in the positive direction. |
| negative | -1    | Travels in the negative direction. |

## 5.7 Velocity Control

| Name     | Value | Description  |
|----------|-------|--|
| shortest | 0     | Not available. Do not specify this.  |
| fastest  | 3     | Not available. Do not specify this.  |
| current  | 2     | Travels to the current direction.<br>Possible to use only for the modulo axis. |

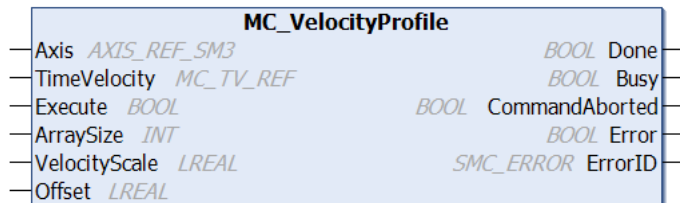
### Info.

- To stop, set Velocity to 0 and execute again.
- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

### 5.7.2 MC\_VelocityProfile (Velocity Profile Movement)

This is a function block (FB) that causes the axis to operate according to the profile data that consists of a combination of time and velocity.

#### ■ Icon



#### ■ Parameter

| Scope          | Name          | Type         | Initial | Description  |
|----------------|---------------|--------------|---------|--|
| Input / output | Axis          | AXIS_REF_SM3 | -       | Specifies the axis.  |
|                | TimeVelocity  | MC_TV_REF    | -       | Specifies the time / velocity profile.   |
| Input          | Execute       | BOOL         | FALSE   | Starts execution at the rising edge.   |
|                | ArraySize     | INT          | 0       | A member of input TimeVelocity. Specifies the number of points to be executed by FB in the array of time and velocity that are specified by MC_TV_ARRAY. |
|                | VelocityScale | LREAL        | 1       | Velocity scaling   |
|                | Offset        | LREAL        | 0       | Velocity offset (u)  |
| Output         | Done          | BOOL         | FALSE   | TRUE: The movements specified by the profile are completed.  |
|                | Busy          | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.  |



| Scope | Name           | Type      | Initial | Description                                       |
|-------|----------------|-----------|---------|---|
|       | CommandAborted | BOOL      | FALSE   | TRUE: An interruption from other FB has occurred. |
|       | Error          | BOOL      | FALSE   | TRUE: An error has occurred within the FB.        |
|       | ErrorID        | SMC_ERROR | 0       | An error ID is output.                            |

#### ■ MC\_TV\_REF (Structure)

| Member          | Type                     | Description  |
|-----------------|--------------------------|--|
| Number_of_pairs | INT                      | Not used   |
| isAbsolute      | BOOL                     | Methods of specifying the velocity of profile data<br>TRUE: Specified in an absolute value.<br>FALSE: Specified in a relative value. |
| MC_TV_Array     | ARRAY [1..100] OF SMC_TV | Time and velocity profile data<br>(1st point to 100th point)   |

#### ■ SMC\_TV (Structure)

| Member     | Type  | Description              |
|------------|-------|--------------------------|
| delta_time | TIME  | Time of the profile data |
| velocity   | LREAL | Velocity of profile data |

Regarding the method for entering defaults for variables of the MC\_TV\_REF Type Structure, refer to “Default Setting for Variables of the MC\_TP\_REF Type Structure”.

#### Note

- While the axis keeps driving, do not specify delta\_time to 0 ms. Otherwise, it may cause an abnormal operation.

#### REFERENCE

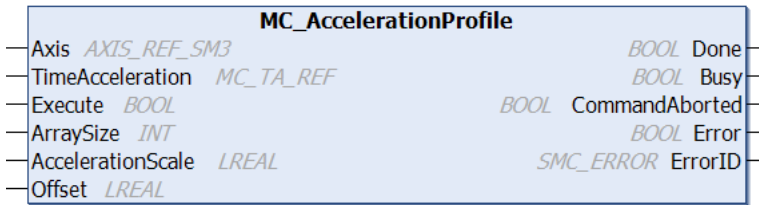
[5.6.6 Default Setting for Variables of the MC\\_TP\\_REF Type Structure](#)

### 5.7.3 MC\_AccelerationProfile (Acceleration Profile Movement)

This is a function block (FB) that causes the axis to operate according to the profile data that consists of a combination of time and acceleration.

## 5.7 Velocity Control

### ■ Icon



### ■ Parameter

| Scope          | Name              | Type         | Initial | Description   |
|----------------|-------------------|--------------|---------|---|
| Input / output | Axis              | AXIS_REF_SM3 | -       | Specifies the axis.   |
|                | TimeAcceleration  | MC_TA_REF    | -       | Specifies the time / acceleration profile.  |
| Input          | Execute           | BOOL         | FALSE   | Starts execution at the rising edge.  |
|                | ArraySize         | INT          | 0       | A member of the input TimeAcceleration.<br>Specifies the number of points to be executed by FB in the array of time and acceleration that are specified by MC_TA_ARRAY. |
|                | AccelerationScale | LREAL        | 1       | Acceleration scaling  |
|                | Offset            | LREAL        | 0       | Acceleration offset (u)   |
| Output         | Done              | BOOL         | FALSE   | TRUE: The movements specified by the profile are completed.   |
|                | Busy              | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.   |
|                | CommandAborted    | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.   |
|                | Error             | BOOL         | FALSE   | TRUE: An error has occurred within the FB.  |
|                | ErrorID           | SMC_ERROR    | 0       | An error ID is output.  |

### ■ MC\_TA\_REF (Structure)

| Member          | Type                     | Description  |
|-----------------|--------------------------|--|
| Number_of_pairs | INT                      | Not used   |
| isAbsolute      | BOOL                     | Methods of specifying the acceleration of profile data<br>TRUE: Specified in an absolute value.<br>FALSE: Specified in a relative value. |
| MC_TA_Array     | ARRAY [1..100] OF SMC_TA | Time and acceleration profile data (1st point to 100th point)  |

### ■ SMC\_TA (Structure)

| Member     | Type | Description              |
|------------|------|--------------------------|
| delta_time | TIME | Time of the profile data |

---

| Member       | Type  | Description                  |
|--------------|-------|------------------------------|
| Acceleration | LREAL | Acceleration of profile data |

Regarding the method for entering defaults for variables of the MC\_TA\_REF Type Structure, refer to “Default Setting for Variables of the MC\_TP\_REF Type Structure”.

 **Note**

- While the axis keeps driving, do not specify delta\_time to 0 ms. Otherwise, it may cause an abnormal operation.

---

**REFERENCE**

[5.6.6 Default Setting for Variables of the MC\\_TP\\_REF Type Structure](#)

---

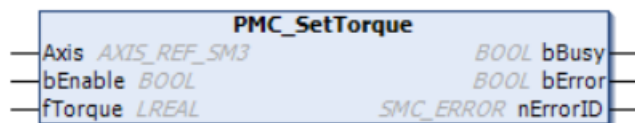
## 5.8 Torque Control

### 5.8 Torque Control

#### 5.8.1 PMC\_SetTorque (Torque Control)

Specifies by % the torque of the axis.

##### ■ Icon



##### ■ Parameter

| Scope          | Name     | Type         | Initial | Description                                 |
|----------------|----------|--------------|---------|---|
| Input / output | Axis     | AXIS_REF_SM3 | -       | Specifies the axis.                         |
| Input          | bEnable  | BOOL         | FALSE   | TRUE: The FB can be executed.               |
|                | fTorque  | LREAL        | 0       | Specifies the ratio (%) to the rated torque |
| Output         | bBusy    | BOOL         | FALSE   | TRUE: Execution of the FB is not completed. |
|                | bError   | BOOL         | FALSE   | TRUE: An error has occurred within the FB.  |
|                | nErrorID | SMC_ERROR    | 0       | An error ID is output.                      |

##### **i** Info.

- This function block can be executed only in the torque control mode (SMC\_torque). Change to the torque control mode in advance using MC\_SetControllerMode.
- Confirm that the MINAS firmware is the latest version.
- When stopping the axis while the axis is being controlled in the torque control mode (SMC\_torque), set fTorque to 0 and execute again. Or, change to the position control mode using SMC\_SetControllerMode and then stop the axis using MC\_Stop.
- When performing torque control for MINAS A5B/A6B, the speed is limited by the Max motorspeed (16#6080:00) set in the PDO mapping. Set the speed in advance.
- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

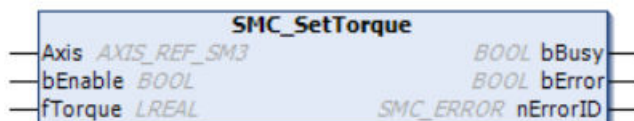
#### REFERENCE

- 5.3.1 SMC\_SetControllerMode (Control Mode Setting)
- 5.4.1 MC\_Stop (Forced Stop)

### 5.8.2 SMC\_SetTorque (Torque Control)

Specifies by Nm the torque of the axis.

#### ■ Icon



#### ■ Parameter

| Scope          | Name     | Type         | Default | Description                                 |
|----------------|----------|--------------|---------|---|
| Input / output | Axis     | AXIS_REF_SM3 | -       | Specifies the axis.                         |
| Input          | bEnable  | BOOL         | FALSE   | TRUE: The FB can be executed.               |
|                | fTorque  | LREAL        | 0       | Specifies the torque (N·m, N)               |
| Output         | bBusy    | BOOL         | FALSE   | TRUE: Execution of the FB is not completed. |
|                | bError   | BOOL         | FALSE   | TRUE: An error has occurred within the FB.  |
|                | nErrorID | SMC_ERROR    | 0       | Error ID output                             |

#### **i** Info.

- This function block can be executed only in the torque control mode (SMC\_torque). Change to the torque control mode in advance using SMC\_SetControllerMode
- Confirm that the MINAS firmware is the latest version.
- When stopping the axis while the axis is being controlled in the torque control mode (SMC\_torque), set fTorque to 0 and execute again. Or, change to the position control mode (SMC\_position) using SMC\_SetControllerMode and then stop the axis using MC\_Stop.
- When performing torque control for MINAS A5B/A6B, the speed is limited by the Max motorspeed (16#6080:00) set in the PDO mapping. Set the speed in advance.
- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

(MEMO)

# 6 Motion Control Function Blocks (Synchronous Control)

---

This section describes motion control function blocks to perform synchronous processing.

|  |      |
|--|------|
| 6.1 Cam Operation.....                                       | 6-2  |
| 6.1.1 MC_CamIn (Start Cam Operation) .....                   | 6-2  |
| 6.1.2 MC_CamOut (Cancel Cam Operation) .....                 | 6-5  |
| 6.1.3 MC_CamTableSelect (Select Cam Table).....              | 6-6  |
| 6.1.4 SMC_GetTappetValue (Tappet Output).....                | 6-8  |
| 6.2 Gear Operation .....                                     | 6-11 |
| 6.2.1 MC_GearIn (Start Gear Operation).....                  | 6-11 |
| 6.2.2 MC_GearInPos (Position Specified Gear Operation) ..... | 6-12 |
| 6.2.3 MC_GearOut (Cancel Gear Operation) .....               | 6-14 |
| 6.3 Phase Correction .....                                   | 6-16 |
| 6.3.1 MC_Phasing (Master Axis Phase Correction) .....        | 6-16 |

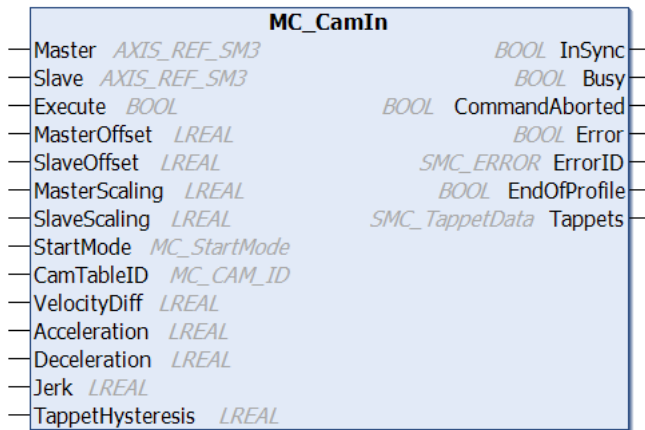
## 6.1 Cam Operation

### 6.1 Cam Operation

#### 6.1.1 MC\_CamIn (Start Cam Operation)

This is a function block (FB) that starts cam synchronous operation. The master axis and the slave axis operate in synchronization according to the cam table.

##### ■ Icon



##### ■ Parameter

| Scope          | Name          | Type         | Initial  | Comment   |
|----------------|---------------|--------------|----------|---|
| Input / output | Master        | AXIS_REF_SM3 | -        | Specifies the master axis.  |
|                | Slave         | AXIS_REF_SM3 | -        | Specifies the slave axis.   |
| Input          | Execute       | BOOL         | FALSE    | Starts execution at the rising edge.  |
|                | MasterOffset  | LREAL        | 0        | Master axis table offset  |
|                | SlaveOffset   | LREAL        | 0        | Slave axis table offset   |
|                | MasterScaling | LREAL        | 1        | Master axis profile scaling factor  |
|                | SlaveScaling  | LREAL        | 1        | Slave axis profile scaling factor   |
|                | StartMode     | MC_StartMode | absolute | Start mode  |
|                | CamTableID    | MC_CAM_ID    | 0        | Cam table ID<br>Specifies the output for MC_CamTableSelect.                       |
|                | VelocityDiff  | LREAL        | 0        | Specifies the maximum velocity difference (u/s) when StartMode is set to ramp_in. |
|                | Acceleration  | LREAL        | 0        | Specifies the acceleration (u/s <sup>2</sup> ) when StartMode is set to ramp_in.  |
|                | Deceleration  | LREAL        | 0        | Specifies the deceleration (u/s <sup>2</sup> ) when StartMode is set to ramp_in.  |

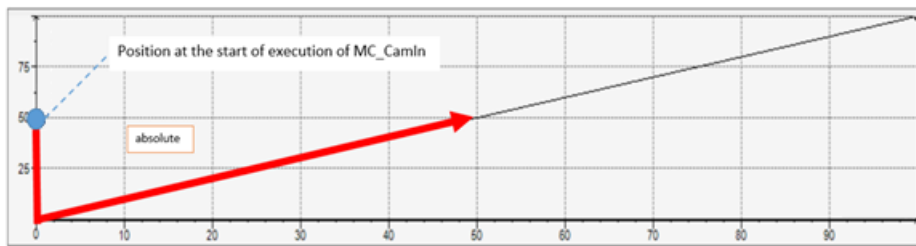


| Scope  | Name             | Type               | Initial | Comment   |
|--------|------------------|--------------------|---------|---|
|        | Jerk             | LREAL              | 0       | Specifies jerk ( $u/s^3$ ) when StartMode is set to ramp_in.                |
|        | TappetHysteresis | LREAL              | 0       | Specifies the hysteresis value (position) of the tappet.                    |
| Output | InSync           | BOOL               | FALSE   | TRUE: The cam is in synchronization for the first time.                     |
|        | Busy             | BOOL               | FALSE   | TRUE: Execution of the FB is not completed.                                 |
|        | CommandAborted   | BOOL               | FALSE   | TRUE: An interruption from other FB has occurred.                           |
|        | Error            | BOOL               | FALSE   | TRUE: An error has occurred within the FB.                                  |
|        | ErrorID          | SMC_ERROR          | 0       | An error ID is output.  |
|        | EndOfProfile     | BOOL               | FALSE   | A pulse is output every time the cam profile period of the slave axis ends. |
|        | Tappets          | SMC_TAPPETD<br>ATA |         | Output tappet data<br>Used as the input for SMC_GetTappetValue.             |

#### ■ MC\_StartMode (Enumeration type)

- absolute

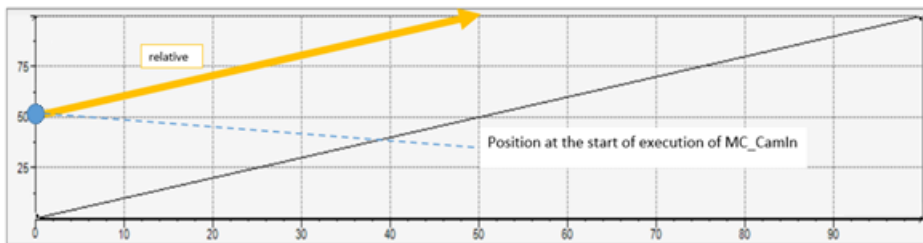
If the slave axis position is offset from the position on the cam table when MC\_CamIn is executed, the position jumps. When starting a new cycle, the cam does not consider the current slave axis position. Therefore, if the slave axis position relative to the mater axis at the start point is offset from the slave axis position relative to the mater axis at the end point, the slave axis position may jump.



- relative

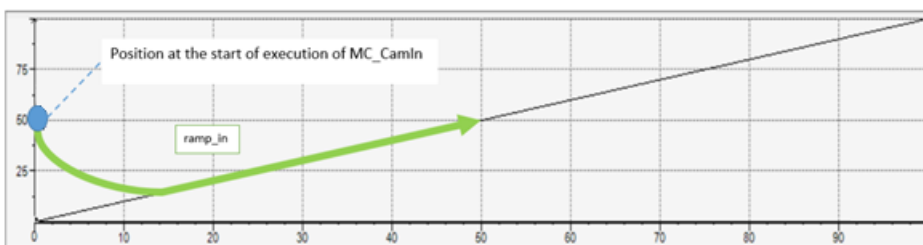
If the slave axis position is offset from the position on the cam table when MC\_CamIn is executed, the position jumps. In consideration of the current slave axis position, new cam operation is started. The slave axis position at the time when the previous cycle is completed is reflected as SlaveOffset on the cam operation. It must be noted, however, that the position may jump if the master axis position is not 0 at the start position.

## 6.1 Cam Operation



- ramp\_in, ramp\_in\_neg, ramp\_in\_pos

When MC\_CamIn is executed, acceleration and deceleration is performed according to VelocityDiff, Acceleration, and Deceleration to reach the synchronized state (InSync). If the slave axis is a modulo axis, correction is made only in the positive direction when the mode is set to ramp\_in\_pos, while correction is made only in the negative direction when the mode is set to ramp\_in\_neg. With the finite axis, ramp\_in\_pos and ramp\_in\_neg are treated as ramp\_in.



The final StartMode is determined by MC\_CamIn.StartMode and MC\_CamTableSelect.SlaveAbsolute.

| MC_CamIn.StartMode | MC_CamTableSelect.SlaveAbsolute | StartMode            |
|--------------------|---------------------------------|----------------------|
| absolute           | TRUE                            | absolute             |
| absolute           | FALSE                           | relative             |
| relative           | TRUE                            | relative             |
| relative           | FALSE                           | relative             |
| ramp_in            | TRUE                            | ramp_in absolute     |
| ramp_in            | FALSE                           | ramp_in relative     |
| ramp_in_pos        | TRUE                            | ramp_in_pos absolute |
| ramp_in_pos        | FALSE                           | ramp_in_pos relative |
| ramp_in_neg        | TRUE                            | ramp_in_neg absolute |
| ramp_in_neg        | FALSE                           | ramp_in_neg relative |

### ■ Changing the scale and offset in the cam table (MasterOffset, SlaveOffset, MasterScaling, SlaveScaling)

A cam table can be created by using the Cam Table Editor on the GM Programmer. The cam table created by using the Cam Table Editor determines the relationship between the master axis position and the slave axis position that operate on the POU,

$$\text{SlavePosition} = \text{CAM}(\text{MasterPosition})$$

The scale and offset in the cam table can be changed on the POU by setting the MC\_CamIn parameters including MasterOffset, SlaveOffset, sterScaling, and SlaveSacing. At that time, the slave axis position is determined as follows according to the setting of each parameter.

$$\text{SlavePosition} = \text{SlaveSacing} * \text{CAM}(\text{MasterPosition} * \text{MasterScaling} + \text{MasterOffset}) + \text{SlaveOffset}$$

■ **Tappet**

For information on setting the tappet and TappetHysteresis of MC\_CamIn, refer to the description on SMC\_GetTappetValue

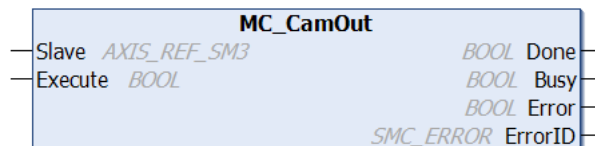
■ **SMC\_TAPPETDATA (Structure)**

| Member      | Type                                     | Description   |
|-------------|--|---|
| pTaps       | ARRAY [0..2] OF POINTER TO SMC_CAMTappet | Used internally for the output of MC_CamIn and for the input of SMC_GetTappetValue. |
| dwCycleTime | DWORD                                    |   |
| byChannels  | BYTE                                     |   |
| bRestart    | BOOL                                     |   |

**6.1.2 MC\_CamOut (Cancel Cam Operation)**

This is a function block (FB) that cancels synchronous operation of the cam. Synchronized operation between the master axis and slave axis is canceled.

■ **Icon**



■ **Parameter**

| Scope          | Name    | Type         | Initial | Description                                      |
|----------------|---------|--------------|---------|--|
| Input / output | Slave   | AXIS_REF_SM3 | -       | Specifies the slave axis to be released.         |
| Input          | Execute | BOOL         | FALSE   | Starts execution at the rising edge.             |
| Output         | Done    | BOOL         | FALSE   | TRUE: Synchronization cancellation is completed. |
|                | Busy    | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.      |
|                | Error   | BOOL         | FALSE   | TRUE: An error has occurred within the FB.       |
|                | ErrorID | SMC_ERROR    | 0       | An error ID is output.                           |

## 6.1 Cam Operation

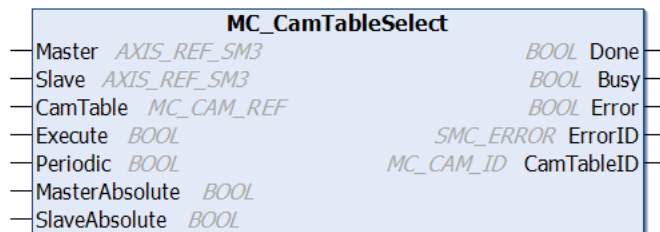
### **i** Info.

- The slave axis operation continues even after the cam operation is canceled. Execute MC\_Halt or MC\_Stop to stop the slave axis.
- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

### 6.1.3 MC\_CamTableSelect (Select Cam Table)

This is a function block (FB) that specifies the cam table to be used for synchronous operation of the cam. When the cam table to be used for synchronized operation between the master axis and slave axis is selected, a cam table ID is output.

#### ■ Icon



#### ■ Parameter

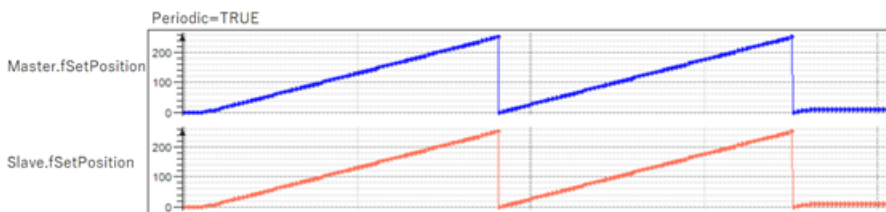
| Scope          | Name           | Type         | Initial | Description   |
|----------------|----------------|--------------|---------|---|
| Input / output | Master         | AXIS_REF_SM3 | -       | Specifies the master axis.  |
|                | Slave          | AXIS_REF_SM3 | -       | Specifies the slave axis.   |
|                | CamTable       | MC_CAM_REF   | -       | Specifies the cam table.  |
| Input          | Execute        | BOOL         | FALSE   | Starts execution at the rising edge.  |
|                | Periodic       | BOOL         | TRUE    | Movement of the slave axis<br>TRUE: Repeat execution<br>FALSE: 1-period execution         |
|                | MasterAbsolute | BOOL         | TRUE    | TRUE: Absolute position of the master axis<br>FALSE: Relative position of the master axis |
|                | SlaveAbsolute  | BOOL         | TRUE    | TRUE: Absolute position of the slave axis<br>FALSE: Relative position of the slave axis   |
| Output         | Done           | BOOL         | FALSE   | TRUE: Selection is completed.   |
|                | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.   |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.  |

| Scope | Name       | Type      | Initial | Description                                     |
|-------|------------|-----------|---------|---|
|       | ErrorID    | SMC_ERROR | 0       | An error ID is output.                          |
|       | CamTableID | MC_CAM_ID |         | Cam table ID<br>Used as the input for MC_CamIn. |

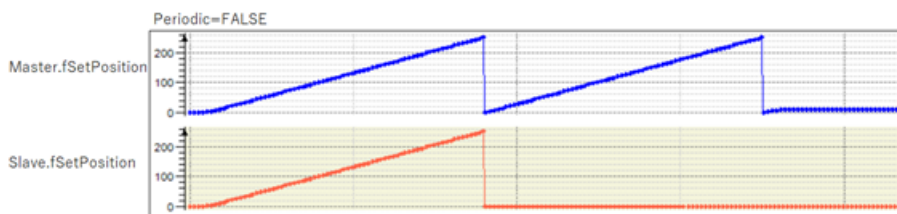
### ■ Periodic (Periodic cam control)

If Periodic of MC\_CamTableSelect is set to TRUE, cam operation is repeatedly performed. The cam is automatically restarted when reaching the end position. If Periodic is set to FALSE, when the master axis reaches the end position, EndOfProfile of MC\_CamIn changes to TRUE and the slave axis stops at the current position. When the master axis enters the cam position range again, the slave axis starts moving according to the cam table.

#### If periodic = TRUE

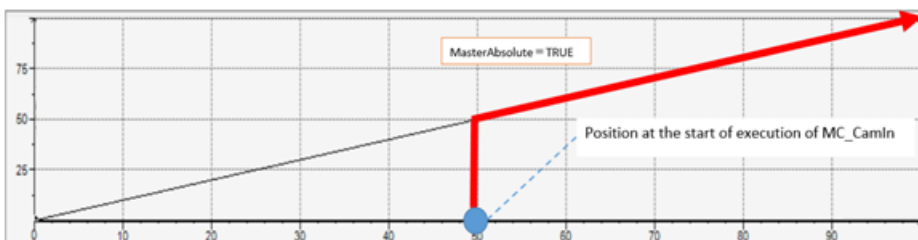


#### If periodic = FALSE



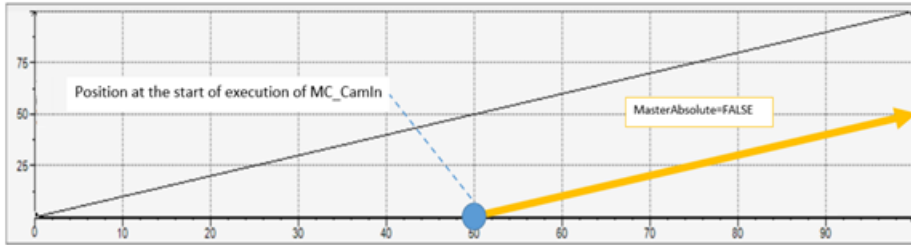
### ■ MasterAbsolute

If MasterAbsolute is set to TRUE, the cam starts from the current master axis position. Any position within the cam table range specified for the master axis can be set as the starting position. If the starting position is outside the cam table range, an error occurs.



If MasterAbsolute is set to FALSE, the cam is relocated to the current master axis position. The zero point of the master axis in the cam table is relocated to the current master axis position. This mode is allowed only when the value 0 is within the master axis range.

## 6.1 Cam Operation



### ■ SlaveAbsolute

Refer to MC\_CamIn.

### **i** Info.

- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

### 6.1.4 SMC\_GetTappetValue (Tappet Output)

This is a function block (FB) that outputs the tappet data set using the cam table. Specify the output Tappets of the MC\_CamIn function block as the input for this FB and obtain the tappet output for one track.

### ■ Icon



### ■ Parameter

| Scope          | Name                 | Type               | Initial | Description  |
|----------------|----------------------|--------------------|---------|--|
| Input / output | Tappets              | SMC_TAPPETD<br>ATA | -       | Specifies the output Tappets of MC_CamIn.  |
| Input          | iID                  | INT                | 0       | Track ID of the tappet   |
|                | bInitValue           | BOOL               | FALSE   | Default of the tappet  |
|                | bSetInitValueAtReset | BOOL               | FALSE   | TRUE: The tappet output is Initialized when MC_CamIn is executed.<br>FALSE: The tappet output is held when MC_CamIn is executed. |
| Output         | bTappet              | BOOL               | FALSE   | Tappet output  |

### ■ SMC\_TAPPETDATA (Structure)

| Member      | Type   | Description   |
|-------------|--|---|
| pTaps       | ARRAY [0..2] OF<br>POINTER TO<br>SMC_CAMTappet | Used internally for the output of MC_CamIn and for the input of SMC_GetTappetValue. |
| dwCycleTime | DWORD  |   |
| byChannels  | BYTE   |   |
| bRestart    | BOOL   |   |

### ■ Tappet settings

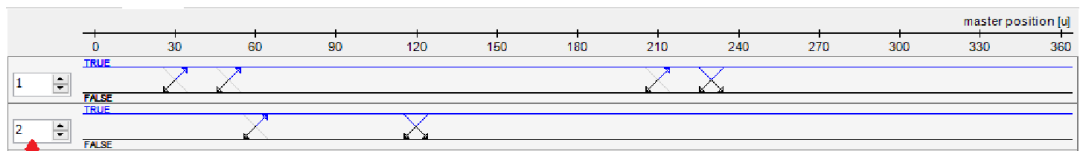
Set tappet settings on the "Tappet" tab or "Tappet Table" tab window.

The settings that are set on these two tab windows are linked to each other. They can be set from either window.

Set switches required for the tracks (100 max).

The following is an example where three switches are set for two tracks.

#### "Tappet"



Track ID

#### "Tappet Table"

|   | Track ID | X   | positive pass | negative pass |              |
|---|----------|-----|---------------|---------------|--------------|
| + | 1        |     |               |               | } Switch (1) |
| ⊖ |          | 30  | switch ON     | switch OFF    |              |
| ⊖ |          | 50  | switch ON     | switch OFF    |              |
| ⊖ | 1        | 210 | switch ON     | switch OFF    | } Switch (2) |
| ⊖ |          | 230 | switch OFF    | switch OFF    |              |
| + | 2        |     |               |               | } Switch (3) |
| ⊖ |          | 60  | switch ON     | switch OFF    |              |
| ⊖ |          | 120 | switch OFF    | switch OFF    |              |

### ■ Operations when the function block is executed

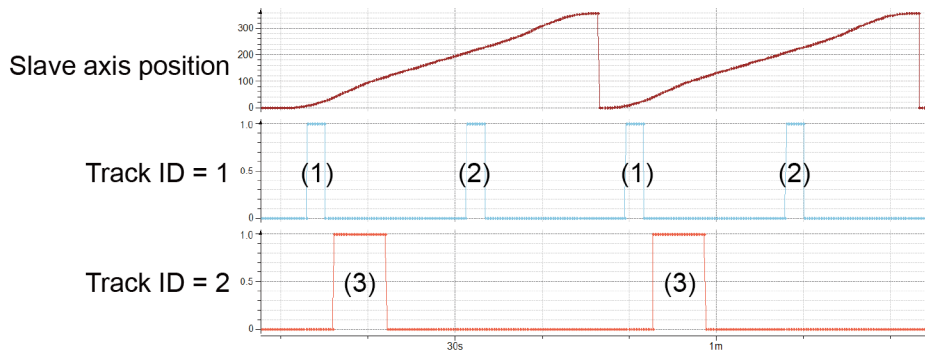
The operations that take place when, after setting the tappet, the SMC\_GetTappetValue function block is executed and the tappet (bTappet) is output. The axis is set to the modulo (modulo value: 360).

#### Trace

(1) to (3) are switch numbers.

## 6.1 Cam Operation

---



### ■ TappetHysteresis of MC\_CamIn

By setting TappetHysteresis, mechanical chattering can be filtered. When the cam reaches the tappet position, tappet processing is performed. Once the cam moves out of the tappet position range set in TappetHysteresis, tappet processing is not performed unless the cam reaches the tappet position again.

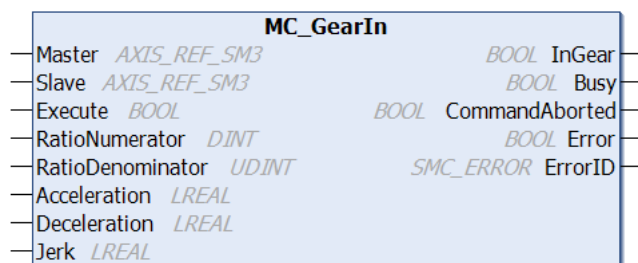


## 6.2 Gear Operation

### 6.2.1 MC\_GearIn (Start Gear Operation)

This is a function block (FB) that starts synchronous operation of the gears. Specify the gear ratio between the master axis and slave axis and start moving the gears for synchronous operation of the gears.

#### ■ Icon



#### ■ Parameter

| Scope          | Name             | Type         | Initial | Description  |
|----------------|------------------|--------------|---------|--|
| Input / output | Master           | AXIS_REF_SM3 | -       | Specifies the master axis.   |
|                | Slave            | AXIS_REF_SM3 | -       | Specifies the slave axis.  |
| Input          | Execute          | BOOL         | FALSE   | Starts execution at the rising edge.                                     |
|                | RatioNumerator   | DINT         | 1       | Specifies the gear ratio (numerator).                                    |
|                | RatioDenominator | UDINT        | 1       | Specifies the gear ratio (denominator).                                  |
|                | Acceleration     | LREAL        | 0       | Maximum acceleration ( $u/s^2$ ) until gear synchronization is completed |
|                | Deceleration     | LREAL        | 0       | Maximum deceleration ( $u/s^2$ ) until gear synchronization is completed |
|                | Jerk             | LREAL        | 0       | Maximum jerk ( $u/s^3$ ) until gear synchronization is completed         |
| Output         | InGear           | BOOL         | FALSE   | TRUE: Gear synchronization is completed.                                 |
|                | Busy             | BOOL         | FALSE   | TRUE: The FB is in operation.  |
|                | CommandAborted   | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.                        |
|                | Error            | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                               |
|                | ErrorID          | SMC_ERROR    | 0       | An error ID is output.   |

## 6.2 Gear Operation

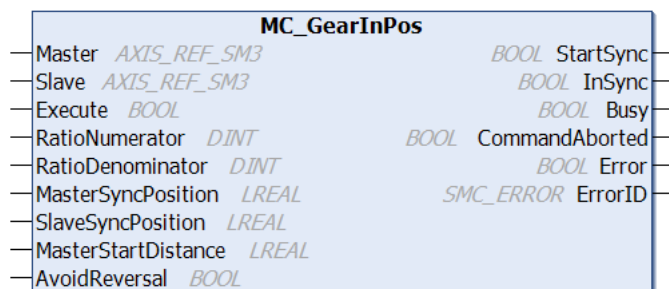
### **i** Info.

- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

### 6.2.2 MC\_GearInPos (Position Specified Gear Operation)

This is a function block (FB) that starts synchronous operation of the gears from the specified absolute position.

#### ■ Icon



#### ■ Parameter

| Scope          | Name                | Type         | Initial | Description   |
|----------------|---------------------|--------------|---------|---|
| Input / output | Master              | AXIS_REF_SM3 | -       | Specifies the master axis.  |
|                | Slave               | AXIS_REF_SM3 | -       | Specifies the slave axis.   |
| Input          | Execute             | BOOL         | FALSE   | Starts execution at the rising edge.  |
|                | RatioNumerator      | DINT         | 1       | Specifies the gear ratio (numerator).   |
|                | RatioDenominator    | DINT         | 1       | Specifies the gear ratio (denominator).   |
|                | MasterSyncPosition  | LREAL        | 0       | Master axis position to start synchronization   |
|                | SlaveSyncPosition   | LREAL        | 0       | Slave axis position to start synchronization  |
|                | MasterStartDistance | LREAL        | 0       | When the master axis moves forward from the MasterSyncPosition position for the distance specified by MasterStartDistance, the slave axis starts moving for the synchronized operation with the master axis.<br>If MasterStartDistance is zero, the slave axis immediately starts moving for synchronization. |
|                | AvoidReversal       | BOOL         | FALSE   | <ul style="list-style-type: none"> <li>• Axis setting: When set to "Finite" AvoidReversal = TRUE</li> </ul>   |

| Scope  | Name           | Type      | Initial | Description  |
|--------|----------------|-----------|---------|--|
|        |                |           |         | <ul style="list-style-type: none"> <li>Axis setting: When set to "Modulo"<br/>AvoidReversal = FALSE</li> </ul> |
| Output | StartSync      | BOOL      | FALSE   | TRUE: Gear synchronization is started.   |
|        | InSync         | BOOL      | FALSE   | TRUE: Gear synchronization is completed.   |
|        | Busy           | BOOL      | FALSE   | TRUE: The FB is in operation.  |
|        | CommandAborted | BOOL      | FALSE   | TRUE: An interruption from other FB has occurred.  |
|        | Error          | BOOL      | FALSE   | TRUE: An error has occurred within the FB.   |
|        | ErrorID        | SMC_ERROR | 0       | An error ID is output.   |

### ■ Operations when the function block is executed

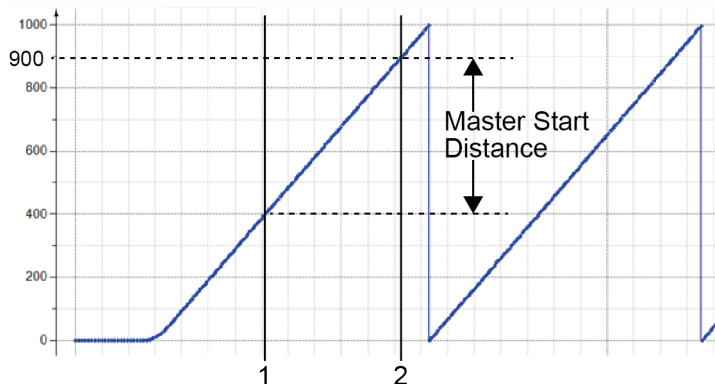
This example shows the trace when the MC\_GearInPos function block is executed with the following conditions.

#### Execution condition

| Item                      | Dis                          |
|---------------------------|------------------------------|
| Master axis type          | Modulo (modulo value = 1000) |
| Slave axis type           | Modulo (modulo value = 1000) |
| Gear ratio                | 1 : 1                        |
| Input MasterSyncPosition  | 900                          |
| Input SlaveSyncPosition   | 900                          |
| Input MasterStartDistance | 500                          |

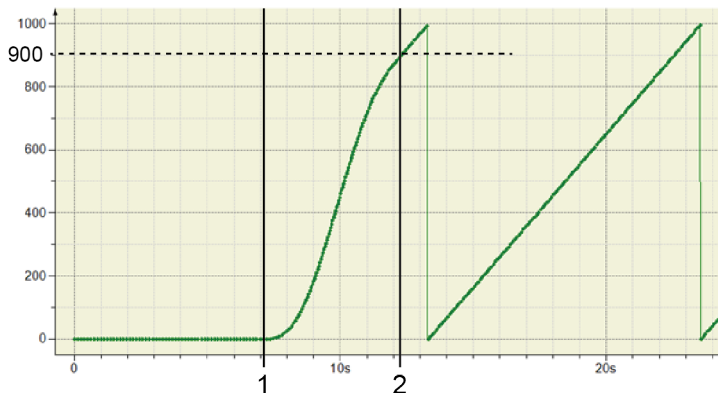
When the master axis position reaches 900 and the slave axis position reaches 900, the master axis starts to synchronize with the slave axis. When the master axis passes the position 400, which is obtained by deducting 500 (MasterStartDistance) from 900 (synchronization start position of the master axis), the slave axis starts traveling to synchronize with the master axis. At this time, velocity, acceleration, and deceleration are automatically determined.

#### Position of the master axis



## 6.2 Gear Operation

### Position of the slave axis



#### ■ AvoidReversal

By setting `AvoidReversal`, the slave axis can be restricted on reverse rotation. If `AvoidReversal` is set to `TRUE`, an error occurs under the following conditions.

1. Gear ratio is negative.
 

If the gear ratio is negative (for example, `RatioNumerator = -1`, `RatioDenominator = 1`), when the axis reaches the position set in `GearInPos.StartSync` while the slave axis is operating in forward rotation, an error (`SMC_GIP_SLAVE_REVERSAL_CANNOT_BE_AVOIDED`) occurs
2. The slave axis is rotating in reverse to the rotation of the master axis before the start of synchronization
 

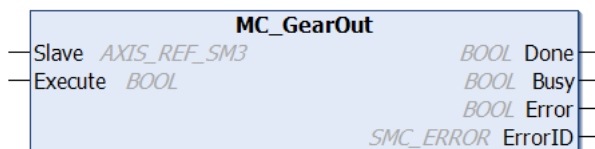
When the axis reaches the gear synchronization start position set in `StartSync` while the slave axis is operating in reverse rotation, an error (`SMC_GIP_SLAVE_REVERSAL_CANNOT_BE_AVOIDED`) occurs
3. Correction of the slave axis is not completed within five cycles.
 

Gear synchronization completion (`InSync`) is not achieved within five cycles after reaching the gear synchronization start (`StartSync`), an error occurs.

### 6.2.3 MC\_GearOut (Cancel Gear Operation)

This is a function block (FB) that cancels synchronous operation of the gears. Synchronized gear operation between the master axis and slave axis is canceled.

#### ■ Icon



### ■ Parameter

| Scope          | Name    | Type         | Initial | Description                                      |
|----------------|---------|--------------|---------|--|
| Input / output | Slave   | AXIS_REF_SM3 | -       | Specifies the axis.                              |
| Input          | Execute | BOOL         | FALSE   | Starts execution at the rising edge.             |
| Output         | Done    | BOOL         | FALSE   | TRUE: Synchronization cancellation is completed. |
|                | Busy    | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.      |
|                | Error   | BOOL         | FALSE   | TRUE: An error has occurred within the FB.       |
|                | ErrorID | SMC_ERROR    | 0       | An error ID is output.                           |

### Info.

- The slave axis operation continues even after the gear operation is canceled. Execute MC\_Halt or MC\_Stop to stop the slave axis.
- Reference manual
  - GM1 Controller RTEX User's Manual (Operation Edition)*
  - GM1 Controller EtherCAT User's Manual (Operation Edition)*

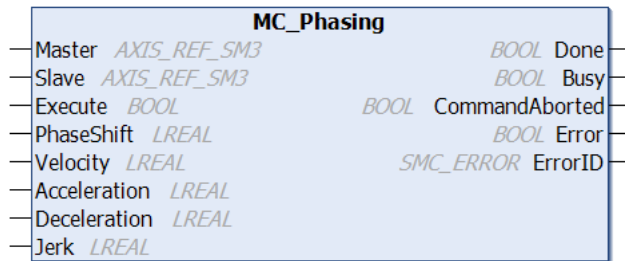
## 6.3 Phase Correction

### 6.3 Phase Correction

#### 6.3.1 MC\_Phasing (Master Axis Phase Correction)

This is a function block (FB) that performs phase correction between the master axis and slave axis. Phase synchronous operation can be performed by making phase correction for the master axis.

##### ■ Icon



##### ■ Parameter

| Scope          | Name           | Type         | Initial | Description   |
|----------------|----------------|--------------|---------|---|
| Input / output | Master         | AXIS_REF_SM3 | -       | Specifies the master axis.                                  |
|                | Slave          | AXIS_REF_SM3 | -       | Specifies the slave axis.                                   |
| Input          | Execute        | BOOL         | FALSE   | Starts execution at the rising edge.                        |
|                | PhaseShift     | LREAL        | 0       | Specifies the phase between the master axis and slave axis. |
|                | Velocity       | LREAL        | 0       | Specifies the velocity (u/s).                               |
|                | Acceleration   | LREAL        | 0       | Specifies the acceleration (u/s <sup>2</sup> ).             |
|                | Deceleration   | LREAL        | 0       | Specifies the deceleration (u/s <sup>2</sup> ).             |
|                | Jerk           | LREAL        | 0       | Specifies the jerk (u/s <sup>3</sup> ).                     |
| Output         | Done           | BOOL         | FALSE   | TRUE: Phase correction is completed.                        |
|                | Busy           | BOOL         | FALSE   | TRUE: The FB is in operation.                               |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.           |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                  |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.                                      |

##### Info.

- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

# 7 Motion Control Function Blocks (Multi-axis Control)

---

This section describes function blocks used to perform multi-axis control using the CNC program.

|  |     |
|--|-----|
| 7.1 Interpolation Control.....                               | 7-2 |
| 7.1.1 PMC_Interpolator2D (2-axis Interpolation Control)..... | 7-2 |
| 7.1.2 PMC_Interpolator3D (3-axis Interpolation Control)..... | 7-3 |
| 7.1.3 PMC_NCDecoder (CNC Table Conversion) .....             | 7-5 |

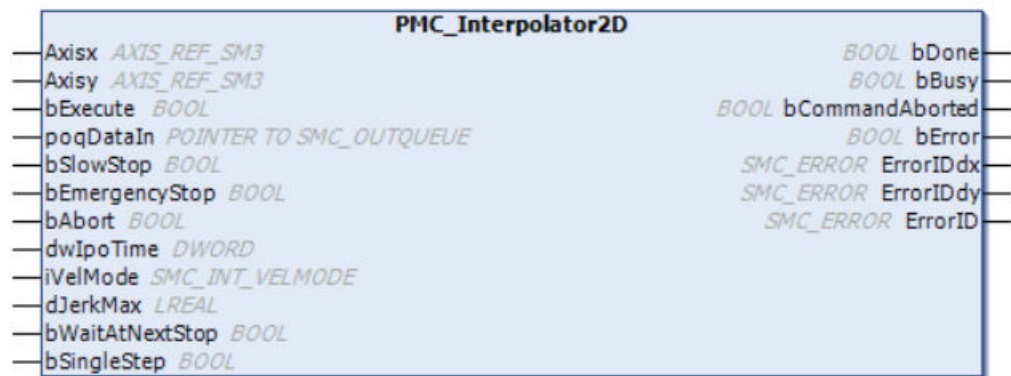
## 7.1 Interpolation Control

### 7.1 Interpolation Control

#### 7.1.1 PMC\_Interpolator2D (2-axis Interpolation Control)

This function block (FB) performs 2-axis interpolation control according to the specified CNC table.

##### ■ Icon



##### ■ Parameter

| Scope | Name           | Type                    | Default value | Description  |
|-------|----------------|-------------------------|---------------|--|
| I/O   | Axisx          | AXIS_REF_SM3            | -             | Specifies the x-axis.  |
|       | Axisy          | AXIS_REF_SM3            | -             | Specifies the y-axis.  |
| Input | bExecute       | BOOL                    | FALSE         | Starts execution at the rising edge.   |
|       | poqDataIn      | POINTER TO SMC_OUTQUEUE | -             | Specifies a pointer to the CNC table.  |
|       | bSlowStop      | BOOL                    | FALSE         | TRUE: A pause is executed. Deceleration stop is executed according to the velocity profile (iVelMode). FALSE: The pause is canceled. |
|       | bEmergencyStop | BOOL                    | FALSE         | TRUE: An emergency stop is executed. FALSE: The emergency stop is canceled.  |
|       | bAbort         | BOOL                    | FALSE         | TRUE: Execution of the FB is stopped.  |
|       | dwIpoTime      | DWORD                   | 0             | MotionTask interval (µsec)   |
|       | iVelMode       | SMC_INT_VELMODE         | TRAPEZOID     | Specifies a velocity profile.  |
|       | dJerkMax       | LREAL                   | LREAL         | Specifies the maximum value of jerk.   |



| Scope  | Name                                | Type      | Default value    | Description   |
|--------|-------------------------------------|-----------|------------------|---|
|        |                                     |           |                  | This parameter must be specified when QUADRATIC is selected for the velocity profile (iVelMode).  |
|        | bWaitAtNextStop <sup>(Note 1)</sup> | BOOL      | BOOL             | TRUE: A pause is executed in the table where the velocity between paths becomes zero.<br>The conditions that cause the velocity between paths to become zero are set in bSingleStep or dAngleMode.<br>FALSE: The pause is canceled. |
|        | bSingleStep <sup>(Note 1)</sup>     | BOOL      | BOOL             | TRUE: All connections between paths are established through deceleration stop.  |
| Output | bCommandAborted                     | BOOL      | FALSE            | TRUE: An interruption is caused by another FB.  |
|        | bBusy                               | BOOL      | -                | TRUE: Execution of the FB is not completed.   |
|        | bDone                               | BOOL      | FALSE            | TRUE: Output is completed.  |
|        | bError                              | BOOL      | FALSE            | TRUE: An error has occurred within the FB.  |
|        | ErrorIDdx                           | SMC_ERROR | SMC_NO_ERR<br>OR | Error ID output during x-axis movement processing   |
|        | ErrorIDdy                           | SMC_ERROR | SMC_NO_ERR<br>OR | Error ID output during y-axis movement processing   |
|        | ErrorID                             | SMC_ERROR | SMC_NO_ERR<br>OR | Error ID output during interpolation control operation  |

(Note 1) When both bWaitAtNextStop and bSingleStep are set to TRUE, they may not work properly, so please do not use them together.

### SMC\_INT\_VELMODE (Enumeration type)

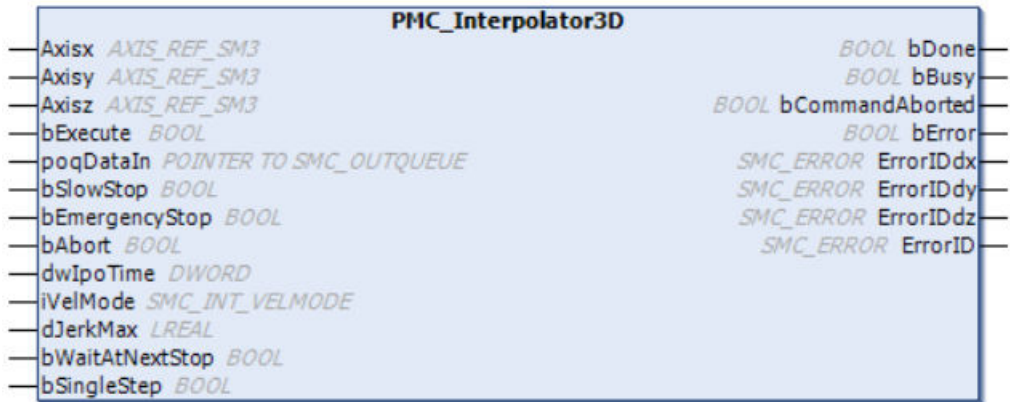
| Name             | Value | Description        |
|------------------|-------|--------------------|
| TRAPEZOID        | 0     | Trapezoid          |
| SIGMOID          | 1     | Sin2               |
| SIGMOID_LIMIT    | 2     | Sin2 (limit)       |
| QUADRATIC        | 3     | Quadratic          |
| QUADRATIC_SMOOTH | 4     | Quadratic (smooth) |

### 7.1.2 PMC\_Interpolator3D (3-axis Interpolation Control)

This function block (FB) performs 3-axis interpolation control according to the specified CNC table.

## 7.1 Interpolation Control

### ■ Icon



### ■ Parameter

| Scope | Name                                | Type                    | Default value | Description  |
|-------|-------------------------------------|-------------------------|---------------|--|
| I/O   | Axisx                               | AXIS_REF_SM3            | -             | Specifies the x-axis.  |
|       | Axisy                               | AXIS_REF_SM3            | -             | Specifies the y-axis.  |
|       | Axisz                               | AXIS_REF_SM3            | -             | Specifies the z-axis.  |
| Input | bExecute                            | BOOL                    | FALSE         | Starts execution at the rising edge.   |
|       | poqDataIn                           | POINTER TO SMC_OUTQUEUE | -             | Specifies a pointer to the CNC table.  |
|       | bSlowStop                           | BOOL                    | FALSE         | TRUE: A pause is executed. Deceleration stop is executed according to the velocity profile (iVelMode). FALSE: The pause is canceled.                 |
|       | bEmergencyStop                      | BOOL                    | FALSE         | TRUE: An emergency stop is executed. FALSE: The emergency stop is canceled.  |
|       | bAbort                              | BOOL                    | FALSE         | TRUE: Execution of the FB is stopped.  |
|       | dwIpoTime                           | DWORD                   | 0             | MotionTask interval (μsec)   |
|       | iVelMode                            | SMC_INT_VELMODE         | TRAPEZOID     | Specifies a velocity profile.  |
|       | dJerkMax                            | LREAL                   | LREAL         | Specifies the maximum value of jerk. This parameter must be specified when QUADRATIC is selected for the velocity profile (iVelMode).                |
|       | bWaitAtNextStop <sup>(Note 1)</sup> | BOOL                    | BOOL          | TRUE: A pause is executed in the table where the velocity between paths becomes zero. The conditions that cause the velocity between paths to become |

| Scope  | Name                            | Type            | Default value | Description  |
|--------|---------------------------------|-----------------|---------------|--|
|        |                                 |                 |               | zero are set in bSingleStep or dAngleMode.<br>FALSE: The pause is canceled.    |
|        | bSingleStep <sup>(Note 1)</sup> | BOOL            | BOOL          | TRUE: All connections between paths are established through deceleration stop. |
| Output | bCommandAborted                 | BOOL            | FALSE         | TRUE: An interruption is caused by another FB.                                 |
|        | bBusy                           | BOOL            | -             | TRUE: Execution of the FB is not completed.                                    |
|        | bDone                           | BOOL            | FALSE         | TRUE: Output is completed.   |
|        | bError                          | BOOL            | FALSE         | TRUE: An error has occurred within the FB.                                     |
|        | ErrorIDdx                       | X-axis error ID | SMC_NO_ERR OR | Error ID output during x-axis movement processing                              |
|        | ErrorIDdy                       | Y-axis error ID | SMC_NO_ERR OR | Error ID output during y-axis movement processing                              |
|        | ErrorIDdz                       | Z-axis error ID | SMC_NO_ERR OR | Error ID output during z-axis movement processing                              |
|        | ErrorID                         | SMC_ERROR       | SMC_NO_ERR OR | Error ID output during interpolation control operation                         |

(Note 1) When both bWaitAtNextStop and bSingleStep are set to TRUE, they may not work properly, so please do not use them together.

### SMC\_INT\_VELMODE (Enumeration type)

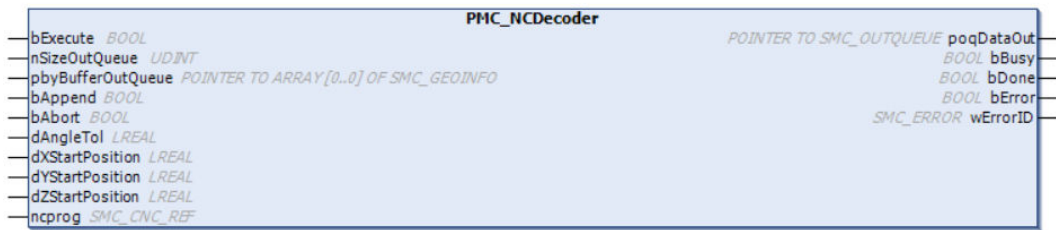
| Name             | Value | Description        |
|------------------|-------|--------------------|
| TRAPEZOID        | 0     | Trapezoid          |
| SIGMOID          | 1     | Sin2               |
| SIGMOID_LIMIT    | 2     | Sin2 (limit)       |
| QUADRATIC        | 3     | Quadratic          |
| QUADRATIC_SMOOTH | 4     | Quadratic (smooth) |

### 7.1.3 PMC\_NCDecoder (CNC Table Conversion)

This function block (FB) decodes the specified SMC\_CNC\_REF value to SMC\_OUTQUEUE.

## 7.1 Interpolation Control

### ■ Icon



### ■ Parameter

| Scope | Name              | Type                                   | Default value | Description   |
|-------|-------------------|--|---------------|---|
| I/O   | ncprogIn          | SMC_CNC_REF                            | -             | Specifies the SMC_CNC_REF value to be decoded.  |
| Input | bExecute          | BOOL                                   | FALSE         | Starts execution at the rising edge.  |
|       | nSizeOutQueue     | UDINT                                  | -             | Specifies a buffer size.<br>We recommend that a buffer be created and the sizeof operator be specified as shown in the following example.<br>ExampleBuf: ARRAY [0..50] OF SMC_GEOINFO;<br>nSizeOutQueue:=sizeof(ExampleBuf)   |
|       | pbyBufferOutQueue | POINTER TO ARRAY [0..0] OF SMC_GEOINFO | -             | Specifies the memory space for SMC_OUTUEUE.<br>We recommend that array SMC_GEOINFO be defined and an address be specified as shown in the following example.<br>ExampleBuf: ARRAY [0..50] OF SMC_GEOINFO;<br>(Buffer that can store 50 path elements)<br>pbyBufferOutQueue:=ADR(ExampleBuf) |
|       | dXstartPosition   | LREAL                                  | 0             | Specifies the position of the x-axis at the start of movement <sup>(Note 1)</sup> .   |
|       | dYstartPosition   | LREAL                                  | 0             | Specifies the position of the y-axis at the start of movement <sup>(Note 1)</sup> .   |
|       | dZstartPosition   | LREAL                                  | 0             | Specifies the position of the z-axis at the start of movement <sup>(Note 1)</sup> .   |
|       | bAppend           | BOOL                                   | FALSE         | TRUE: Decoded data of ncprogIn is appended to the end of poqDataOut without resetting the poqDataOut data within the FB at the rising edge as specified by bExecute.  |
|       | bAbort            | BOOL                                   | FALSE         | TRUE: Execution of the FB is stopped.   |

| Scope  | Name       | Type                    | Default value | Description   |
|--------|------------|-------------------------|---------------|---|
| Output | poqDataOut | POINTER TO SMC_OUTQUEUE | -             | Pointer to SMC_OUTQUEUE which manages decoded SMC_GEOINFO objects |
|        | Busy       | BOOL                    | FALSE         | TRUE: Execution of the FB is not completed.                       |
|        | Done       | BOOL                    | FALSE         | TRUE: Output is completed.  |
|        | Error      | BOOL                    | FALSE         | TRUE: An error has occurred within the FB.                        |
|        | ErrorID    | SMC_ERROR               | 0             | Error ID output   |

(Note 1) We recommend that fSetPosition be entered. If the entered value and the actual position differ, there is a risk that the axis may move suddenly.

(MEMO)

# 8 Motion Control Function Blocks (Motion Communication Control)

This section describes function blocks used to perform communication control.

|  |      |
|--|------|
| 8.1 RTEX .....   | 8-2  |
| 8.1.1 Types of Data To Be Handled by AMP Function Blocks .....                               | 8-2  |
| 8.1.2 RTEX_ClearAmpAlarm (Clear Amplifier Alarm).....  | 8-2  |
| 8.1.3 RTEX_ReadAmpAlarm (Read Amplifier Alarm).....  | 8-4  |
| 8.1.4 RTEX_ReadAmpState (Amplifier Alarm Status) .....                                       | 8-5  |
| 8.1.5 RTEX_ReadAmpData (Amplifier Monitor).....  | 8-6  |
| 8.1.6 RTEX_ReadAmpParameter (Read Amplifier Parameter) .....                                 | 8-7  |
| 8.1.7 RTEX_WriteAmpParameter (Write Amplifier Parameter) .....                               | 8-8  |
| 8.1.8 RTEX_WriteAmpEEPROM (Write Amplifier EEPROM).....                                      | 8-9  |
| 8.1.9 RTEX_Reset (Reset RTEX).....   | 8-9  |
| 8.1.10 RTEX_ClearAmpMultiTurnData (Clear Amplifier Multi-turn Data) .                        | 8-10 |
| 8.1.11 RTEX_ClearAmpPositionalDeviation (Clear Amplifier Deviation<br>Counter).....          | 8-11 |
| 8.1.12 RTEX_GetTrackingCommandError (Read RTEX Command Send<br>Statistics Information) ..... | 8-12 |
| 8.1.13 RTEX_ReadPot (Read POT of Amplifier) .....  | 8-13 |
| 8.1.14 RTEX_ReadNot (Read NOT of Amplifier).....   | 8-14 |
| 8.2 EtherCAT .....   | 8-15 |
| 8.2.1 ETC_CO_SdoRead (Read Slave Parameter).....   | 8-15 |
| 8.2.2 ETC_CO_SdoRead4 (Read Four Bytes of Slave Parameter) .....                             | 8-16 |
| 8.2.3 ETC_CO_SdoReadDWord (Read Double Word of Slave<br>Parameter).....                      | 8-17 |
| 8.2.4 ETC_CO_SdoRead_Access (Read Slave Parameter Index) .....                               | 8-18 |
| 8.2.5 ETC_CO_SdoRead_Channel (Read Priority Specification of Slave<br>Parameter).....        | 8-19 |
| 8.2.6 ETC_CO_SdoWrite (Write Slave Parameter) .....  | 8-21 |
| 8.2.7 ETC_CO_SdoWrite4 (Write Four Bytes of Slave Parameter).....                            | 8-22 |
| 8.2.8 ETC_CO_SdoWriteDWord (Write Double Words of Slave<br>Parameter).....                   | 8-24 |
| 8.2.9 ETC_CO_SdoWrite_Access (Write Slave Parameter Index).....                              | 8-25 |
| 8.2.10 ReadIdentification (Read Slave Identification Data) .....                             | 8-26 |
| 8.2.11 ReadMemory (Read Slave Memory).....   | 8-27 |
| 8.2.12 ReadNbrSlaves (Read the Number of Connected Slaves).....                              | 8-28 |
| 8.2.13 WriteMemory (Write Slave Memory) .....  | 8-29 |
| 8.2.14 PETC_ClearAmpPositionalDeviation (Clear Amplifier Deviation<br>Counter).....          | 8-30 |
| 8.2.15 IoDrvEtherCAT.GetStatistics (Get EtherCAT Frame Statistics<br>Information).....       | 8-31 |

## 8.1 RTEX

### 8.1 RTEX

#### 8.1.1 Types of Data To Be Handled by AMP Function Blocks

| Item              | Description  | Related function blocks                         |
|-------------------|--|---|
| AMP alarm         | This is an AMP alarm that occurs in AMP operation.   | RTEX_ClearAmpAlarm<br>RTEX_ReadAmpAlarm         |
| AMP warning       | This is an AMP warning that occurs in AMP operation.<br>This occurs before the AMP alarm. If the situation worsens, an AMP alarm occurs.   | RTEX_ReadAmpState                               |
| Monitor data      | This is monitor data (position deviation, load percentage, etc.) of the RTEX communication data.   | RTEX_ReadAmpData                                |
| AMP parameter     | This is configuration data of the AMP device itself.   | RTEX_ReadAmpParameter<br>RTEX_WriteAmpParameter |
| Multi-turn data   | There are two types of data in the data read by the absolute encoder (23 bit/r): one type is single-turn data that indicates the position within one motor rotation and the other is multi-turn data that counts one for one turn.   | RTEX_ClearAmpMultiTurnData                      |
| Deviation counter | This is a processing part in the AMP that receives move commands to the AMP.<br>The motor moves according to the commands accumulated in the deviation counter.<br>The commands used for the motor movement are deleted from the deviation counter.<br>The amount of commands accumulated in the deviation counter is called the position deviation. | RTEX_ClearAmpPositionalDeviation                |
| Limit switch      | This data is collected to monitor the POT and NOT states of the AMP.   | RTEX_ReadNot, RTEX_ReadPot                      |

#### 8.1.2 RTEX\_ClearAmpAlarm (Clear Amplifier Alarm)

This is a function block (FB) that clears the AMP alarm. It deletes the alarm or warning that has occurred in the AMP.



### ■ Icon



### ■ Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / Output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute        | BOOL         | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
| Output         | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.                      |
|                | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                            |
|                | Done           | BOOL         | FALSE   | TRUE: Clearing is completed.   |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                             |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.   |
|                | Alarm          | ALARM_CODE   | -       | A deleted alarm code is output.  |
|                | Warning        | WARNING_CODE | -       | A deleted warning code is output.                                      |

### ■ ALARM\_CODE (Union)

| Member           | Type                | Description                        |
|------------------|---------------------|------------------------------------|
| uiAlarmCode      | UINT                | Alarm code                         |
| tAlarmCodeMember | ALARM_WARNING_CODES | Main alarm code and sub alarm code |

### ■ WARNING\_CODE (Union)

| Member             | Type                | Description   |
|--------------------|---------------------|---|
| uiWarningCode      | UINT                | Warning code  |
| tWarningCodeMember | ALARM_WARNING_CODES | Main warning code (warning number) and sub warning code (0) |

## 8.1 RTEX

### ■ ALARM\_WARNING\_CODES (Structure)

| Member     | Type | Description |
|------------|------|-------------|
| byMainCode | BYTE | Main code   |
| bySubCode  | BYTE | Sub code    |

### REFERENCE

[12.2.1 RTEX Error ID](#)

[12.2.2 Alarm Codes](#)

[12.2.3 Warning Codes](#)

### 8.1.3 RTEX\_ReadAmpAlarm (Read Amplifier Alarm)

This is a function block (FB) that reads the AMP alarm. It reads the information of the alarm or warning that has occurred in the AMP.

### ■ Icon



### ■ Parameter

| Scope          | Name           | Type         | Initial | Description   |
|----------------|----------------|--------------|---------|---|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.   |
| Input          | Execute        | BOOL         | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing.        |
|                | AlarmIndex     | UINT         | -       | Specifies the history number (0 to 14).<br>0 is given for the latest history. |
| Output         | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.                             |
|                | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                                   |
|                | Done           | BOOL         | FALSE   | TRUE: Reading is completed.   |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                                    |

| Scope | Name    | Type         | Initial | Description                    |
|-------|---------|--------------|---------|--------------------------------|
|       | ErrorID | SMC_ERROR    | 0       | An error ID is output.         |
|       | Alarm   | ALARM_CODE   | -       | A read alarm code is output.   |
|       | Warning | WARNING_CODE | -       | A read warning code is output. |

#### ■ ALARM\_CODE (Union)

| Member           | Type                | Description                        |
|------------------|---------------------|------------------------------------|
| uiAlarmCode      | UINT                | Alarm code                         |
| tAlarmCodeMember | ALARM_WARNING_CODES | Main alarm code and sub alarm code |

#### ■ WARNING\_CODE (Union)

| Member             | Type                | Description                            |
|--------------------|---------------------|--|
| uiWarningCode      | UINT                | Warning code                           |
| tWarningCodeMember | ALARM_WARNING_CODES | Main warning code and sub warning code |

#### ■ ALARM\_WARNING\_CODES (Structure)

| Member     | Type | Description            |
|------------|------|------------------------|
| byMainCode | BYTE | Main alarm number code |
| bySubCode  | BYTE | Sub alarm number code  |

### REFERENCE

[12.2.1 RTEX Error ID](#)

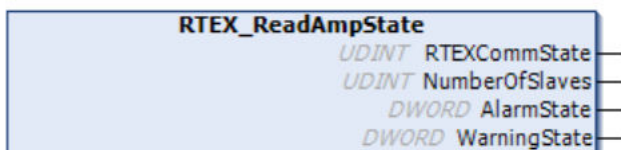
[12.2.2 Alarm Codes](#)

[12.2.3 Warning Codes](#)

#### 8.1.4 RTEX\_ReadAmpState (Amplifier Alarm Status)

This is a function block (FB) that reads the AMP alarm state. It outputs the information and state of the axis where the AMP alarm or warning has occurred.

#### ■ Icon



## 8.1 RTEX

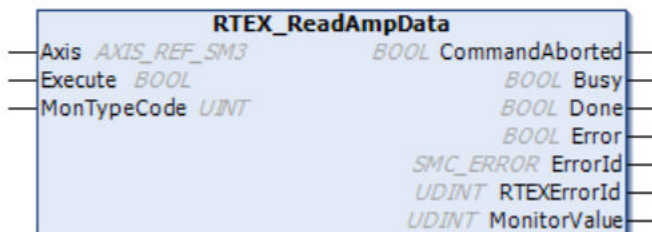
### ■ Parameter

| Scope  | Name           | Type  | Initial | Description   |
|--------|----------------|-------|---------|---|
| Output | RTEXCommState  | UDINT | -       | Outputs RTEX communication status<br>0: Initialization in progress<br>100: Operation in progress<br>1000 or more: Error |
|        | NumberOfSlaves | UDINT | -       | The number of axes connected (1 to 16) is output.   |
|        | AlarmState     | DWORD | -       | The MAC-ID (0 to 15) where the AMP alarm has occurred is output.  |
|        | WarningState   | DWORD | -       | The MAC-ID (0 to 15) where the AMP warning has occurred is output.  |

### 8.1.5 RTEX\_ReadAmpData (Amplifier Monitor)

This is a function block (FB) that reads the monitor data of the AMP. It reads various monitor data of the AMP.

### ■ Icon



### ■ Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute        | BOOL         | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
|                | MonTypeCode    | UINT         |         | Specifies the type code for the monitor command.                       |
| Output         | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.                      |
|                | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                            |
|                | Done           | BOOL         | FALSE   | TRUE: Monitoring is completed.   |

| Scope | Name         | Type      | Initial | Description                                |
|-------|--------------|-----------|---------|--|
|       | Error        | BOOL      | FALSE   | TRUE: An error has occurred within the FB. |
|       | ErrorID      | SMC_ERROR | 0       | An error ID is output.                     |
|       | MonitorValue | UDINT     | -       | Read monitor command                       |

---

**REFERENCE**


---

[12.2.1 RTEX Error ID](#)

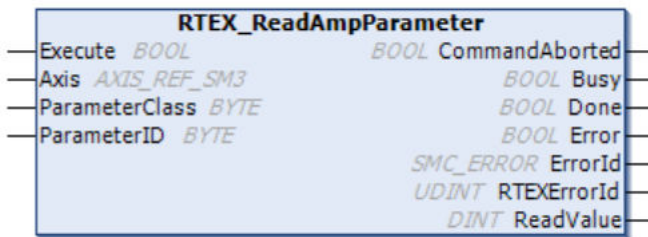
[12.4 Monitor Commands](#)

---

### 8.1.6 RTEX\_ReadAmpParameter (Read Amplifier Parameter)

This is a function block (FB) that reads the AMP parameter.

■ **Icon**



■ **Parameter**

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute        | BOOL         | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
|                | ParameterClass | BYTE         | -       | Specifies the AMP parameter classification.                            |
|                | ParameterID    | BYTE         | -       | Specifies the AMP parameter number.                                    |
| Output         | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.                      |
|                | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                            |
|                | Done           | BOOL         | FALSE   | TRUE: Reading is completed.  |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                             |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.   |

## 8.1 RTEX

| Scope | Name      | Type | Initial | Description              |
|-------|-----------|------|---------|--------------------------|
|       | ReadValue | DINT | -       | Read AMP parameter value |

### REFERENCE

[12.2.1 RTEX Error ID](#)

[12.3 List of AMP Parameters](#)

### 8.1.7 RTEX\_WriteAmpParameter (Write Amplifier Parameter)

This is a function block (FB) that writes the AMP parameter.

#### Icon



#### Parameter

| Scope          | Name           | Type         | Initial | Description  |
|----------------|----------------|--------------|---------|--|
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute        | BOOL         | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
|                | ParameterClass | BYTE         | -       | Specifies the AMP parameter classification.                            |
|                | ParameterID    | BYTE         | -       | Specifies the AMP parameter number.                                    |
|                | WriteValue     | DINT         | -       | Value to be written in the AMP parameter                               |
| Output         | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.                      |
|                | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                            |
|                | Done           | BOOL         | FALSE   | TRUE: Writing is completed.  |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                             |
|                | ErrorID        | SMC_ERROR    | 0       | An error ID is output.   |

### REFERENCE

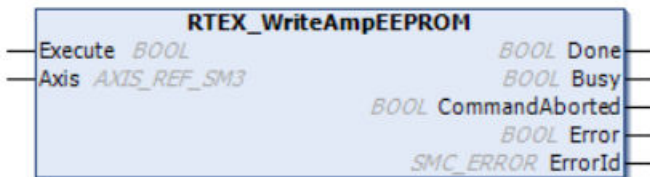
[12.2.1 RTEX Error ID](#)

12.3 List of AMP Parameters

**8.1.8 RTEX\_WriteAmpEEPROM (Write Amplifier EEPROM)**

This is a function block (FB) that writes the servo amplifier parameters to EEPROM.

■ **Icon**



■ **Parameter**

| Scope          | Name           | Type         | Default | Description  |
|----------------|----------------|--------------|---------|--|
| Input          | Execute        | BOOL         | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
| Input / output | Axis           | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Output         | Done           | BOOL         | FALSE   | TRUE: Writing is completed.  |
|                | Busy           | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                            |
|                | CommandAborted | BOOL         | FALSE   | TRUE: An interruption from other FB has occurred.                      |
|                | Error          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                             |
|                | ErrorID        | SMC_ERROR    | 0       | Error ID output  |

**8.1.9 RTEX\_Reset (Reset RTEX)**

Resets the entire RTEX network.

■ **Icon**



## 8.1 RTEX

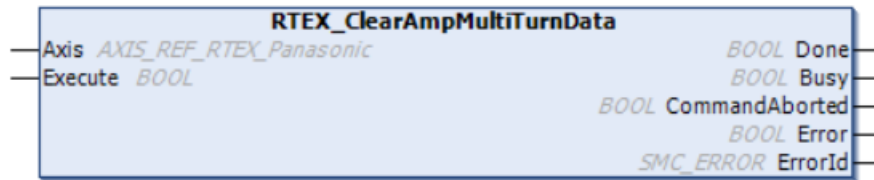
### ■ Parameter

| Scope          | Name           | Type      | Default | Description                                       |
|----------------|----------------|-----------|---------|---|
| Input / output | Execute        | BOOL      | FALSE   | TRUE: Starts execution at the rising edge.        |
| Output         | Done           | BOOL      | FALSE   | TRUE: Reset done                                  |
|                | Busy           | BOOL      | FALSE   | TRUE: Execution of the FB is not completed.       |
|                | CommandAborted | BOOL      | FALSE   | TRUE: An interruption from other FB has occurred. |
|                | Error          | BOOL      | FALSE   | TRUE: An error has occurred within the FB.        |
|                | ErrorID        | SMC_ERROR | 0       | Error ID output                                   |

### 8.1.10 RTEX\_ClearAmpMultiTurnData (Clear Amplifier Multi-turn Data)

This is a function block (FB) that clears the multi-turn data of the AMP.

### ■ Icon



### ■ Parameter

| Scope          | Name           | Type                        | Initial | Description  |
|----------------|----------------|-----------------------------|---------|--|
| Input / output | Axis           | AXIS_REF_RTE<br>X_Panasonic | -       | Specifies the axis.  |
| Input          | Execute        | BOOL                        | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
| Output         | CommandAborted | BOOL                        | FALSE   | TRUE: An interruption from other FB has occurred.                      |
|                | Busy           | BOOL                        | FALSE   | TRUE: Execution of the FB is not completed.                            |
|                | Done           | BOOL                        | FALSE   | TRUE: Clearing is completed.   |
|                | Error          | BOOL                        | FALSE   | TRUE: An error has occurred within the FB.                             |
|                | ErrorID        | SMC_ERROR                   | 0       | An error ID is output.   |

The PMC\_ClearAmpMultiTurnData function block outputs the following errors.

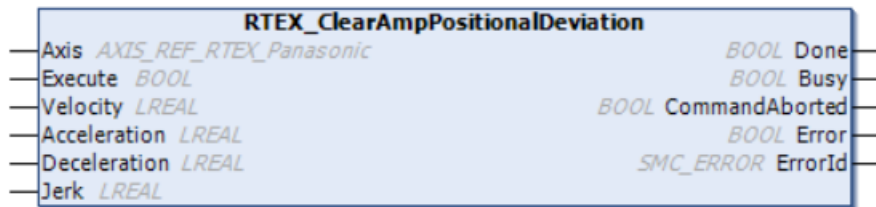


| Error                                 | Description  |
|---------------------------------------|--|
| SMC_WRONG_CONTROLLER_MODE             | Executed in a mode other than the position control mode.<br>Change to SMC_position using SMC_SetControllerMode.                        |
| SMC_DI_HOMING_ERROR                   | The encoder used is an Incremental encoder.  |
| SMC_AXIS_NOT_READY_FOR_MOTION         | The axis is in a state where RTEX_ClearAmpMultiTurnData cannot be executed. It can be executed only when set to Disabled or Errorstop. |
| SMC_REGULATOR_OR_START_NOT_SET        | The axis is in a servo ON state.   |
| SMC_AXIS_REF_CHANGED_DURING_OPERATION | The Axis was changed during operation.   |

### 8.1.11 RTEX\_ClearAmpPositionalDeviation (Clear Amplifier Deviation Counter)

This is a function block (FB) that clears the deviation counter of the AMP. It deletes the position deviation data in the deviation counter of the AMP.

#### ■ Icon



#### ■ Parameter

| Scope          | Name                             | Type                     | Initial | Description                                       |
|----------------|----------------------------------|--------------------------|---------|---|
| Input / output | Axis                             | AXIS_REF_RTE X_Panasonic | -       | Specifies the axis.                               |
| Input          | Execute                          | BOOL                     | FALSE   | Starts execution at the rising edge.              |
|                | Velocity <sup>(Note 1)</sup>     | LREAL                    |         | Information required to execute MC_MoveAbsolute   |
|                | Acceleration <sup>(Note 1)</sup> | LREAL                    |         | Information required to execute MC_MoveAbsolute   |
|                | Deceleration <sup>(Note 1)</sup> | LREAL                    |         | Information required to execute MC_MoveAbsolute   |
|                | Jerk <sup>(Note 1)</sup>         | LREAL                    |         | Information required to execute MC_MoveAbsolute   |
| Output         | CommandAborted                   | BOOL                     | FALSE   | TRUE: An interruption from other FB has occurred. |

## 8.1 RTEX

| Scope | Name    | Type      | Initial | Description                                 |
|-------|---------|-----------|---------|---|
|       | Busy    | BOOL      | FALSE   | TRUE: Execution of the FB is not completed. |
|       | Done    | BOOL      | FALSE   | TRUE: Clearing is completed.                |
|       | Error   | BOOL      | FALSE   | TRUE: An error has occurred within the FB.  |
|       | ErrorID | SMC_ERROR | 0       | An error ID is output.                      |

(Note 1) This function block internally substitutes the command position with an actual position to call MC\_MoveAbsolute and, therefore, requires parameters including Velocity, Acceleration, Deceleration, and Jerk.

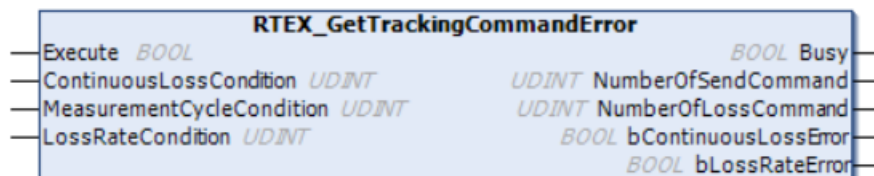
The RTEX\_ClearAmpPositionalDeviation function block outputs the following errors.

| Error                          | Description   |
|--------------------------------|---|
| SMC_WRONG_CONTROLLER_MODE      | Executed in a mode other than the position control mode.<br>Change to SMC_position using SMC_SetControllerMode.                       |
| SMC_AXIS_NOT_READY_FOR_MOTION  | The axis is in a state where RTEX_ClearAmpPositionalDeviation cannot be executed.<br>It can be executed only at the standstill state. |
| SMC_REGULATOR_OR_START_NOT_SET | The axis is in a servo ON state.  |
| SMC_PP_WRONG_AXIS_TYPE         | The axis is a virtual axis.   |

### 8.1.12 RTEX\_GetTrackingCommandError (Read RTEX Command Send Statistics Information)

The RTEX periodically sends commands. With the GM1 specifications, when the MotionTask cycle time exceeds the control cycle, the command position for the servo amplifier is not updated for that cycle. (This is called a lost RTEX command.) This function block measures the number of sent RTEX commands and the number of lost RTEX commands. Using this function, you can check if the command position is updated normally for every cycle.

#### ■ Icon



#### ■ Parameter

| Scope | Definition | Value | Description  |
|-------|------------|-------|--|
| Input | Execute    | ---   | Execute = TRUE: Starts measurement when triggered. |

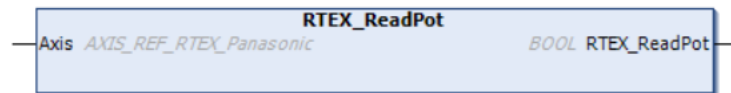
| Scope  |                          | Definition   | Value       | Description   |
|--------|--------------------------|--|-------------|---|
|        |                          |  |             | Execute = FALSE: Clears output.   |
|        | ContinuousLossCondition  | Continuous command loss condition                    | 0: Disabled | If the command loss continuously occurs at ContinuousLossCondition cycle, bContinuousLossError turns TRUE.  |
|        | MesurementCycleCondition | Command loss statistical measurement cycle condition | 0: Disabled | If the command loss occurs as many times as specified in LossRateCondition during the MesurementCycleCondition period, bLossRateError turns TRUE. |
|        | LossRateCondition        | Under measurement                                    | 0 to 100%   |   |
| Output | Busy                     | Under measurement                                    |             | ---   |
|        | NumberOfSendCommand      | Total number of commands sent                        |             | Returns a value when Execute is TRUE. Clears when Execute is FALSE.   |
|        | NumberOfLossCommand      | Total number of commands lost                        |             | Returns a value when Execute is TRUE. Clears when Execute is FALSE.   |
|        | bContinuousLossError     | Occurrence of a continuous command loss error        |             | Occurrence of a condition error of ContinuousLossCondition  |
|        | bLossRateError           | Occurrence of a command loss statistics error        |             | Occurrence of a condition error of MesurementCycleCondition or LossRateCondition  |

(Note 1) If the number of frames exceeds 32 bits, normal value is not returned.

### 8.1.13 RTEX\_ReadPot (Read POT of Amplifier)

This is a function that reads the POT state of the amplifier.

#### ■ Icon



#### ■ Parameter

| Type   | Parameter name | Type                    | Description         |
|--------|----------------|-------------------------|---------------------|
| I/O    | Axis           | AXIS_REF_RTEX_Panasonic | Specifies the axis. |
| Output | RTEX_ReadPot   | BOOL                    | TRUE: POT is ON.    |

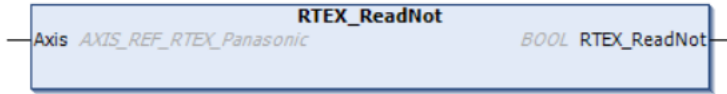
## 8.1 RTEX

---

### 8.1.14 RTEX\_ReadNot (Read NOT of Amplifier)

This is a function that reads the NOT state of the amplifier.

#### ■ Icon



#### ■ Parameter

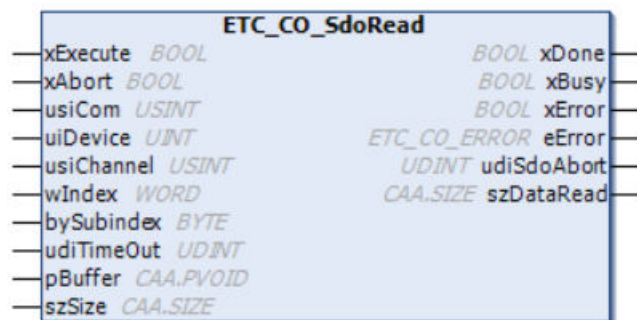
| Type   | Parameter name | Type                    | Description         |
|--------|----------------|-------------------------|---------------------|
| I/O    | Axis           | AXIS_REF_RTEX_Panasonic | Specifies the axis. |
| Output | RTEX_ReadNot   | BOOL                    | TRUE: NOT is ON.    |

## 8.2 EtherCAT

### 8.2.1 ETC\_CO\_SdoRead (Read Slave Parameter)

This is a function block (FB) that reads the EtherCAT slave parameters. Unlike ETC\_CO\_SdoRead4, this FB supports parameters longer than 4 bytes. Specify parameters to be read using the index and sub-index used for the object directory.

#### ■ Icon



#### ■ Parameter

| Scope  | Name       | Type      | Default | Description  |
|--------|------------|-----------|---------|--|
| Input  | xExecute   | BOOL      | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|        | xAbort     | BOOL      | FALSE   | TRUE: Interrupts processing and resets output.   |
|        | usiCom     | USINT     | 1       | 1 (Fixed)  |
|        | uiDevice   | UINT      | 0       | Physical slave address   |
|        | usiChannel | USINT     | 1       | Reserved   |
|        | wIndex     | WORD      | 0       | Parameter index in the object directory <sup>(Note 1)</sup>                                      |
|        | bySubIndex | BYTE      | 0       | Parameter sub-index in the object directory <sup>(Note 1)</sup>                                  |
|        | udiTimeout | UDINT     | 0       | Timeout (Unit: ms)   |
|        | pBuffer    | CAA.PVOID | 0       | Pointer to the buffer that stores read data  |
|        | szSize     | CAA.SIZE  | 0       | Size of the buffer that stores data  |
| Output | xDone      | BOOL      | FALSE   | TRUE: FB processing is completed.  |
|        | xBusy      | BOOL      | FALSE   | TRUE: FB is in progress.   |
|        | xError     | BOOL      | FALSE   | TRUE: An error has occurred within the FB.   |

## 8.2 EtherCAT

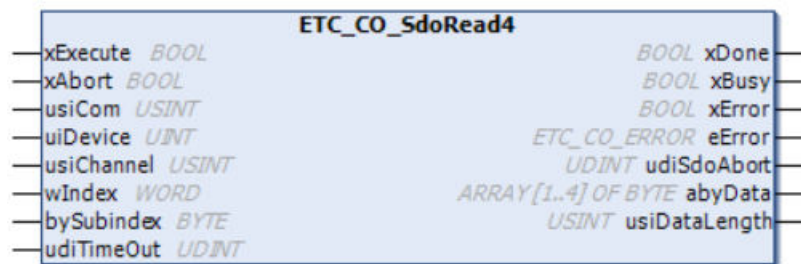
| Scope | Name        | Type         | Default         | Description                               |
|-------|-------------|--------------|-----------------|---|
|       | eError      | ETC_CO_ERROR | ETC_CO_NO_ERROR | Error ID output                           |
|       | udiSdoAbort | UDINT        | 0               | Abort code received from the slave device |
|       | szDataRead  | CAA.SIZE     | 0               | Number of bytes read normally             |

(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

### 8.2.2 ETC\_CO\_SdoRead4 (Read Four Bytes of Slave Parameter)

This is a function block (FB) that reads the EtherCAT slave parameters. Unlike ETC\_CO\_SdoRead, this FB supports only parameters with 4 bytes or less. Specify parameters to be read using the index and sub-index used for the object directory.

#### ■ Icon



#### ■ Parameter

| Scope  | Name       | Type  | Default | Description  |
|--------|------------|-------|---------|--|
| Input  | xExecute   | BOOL  | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|        | xAbsort    | BOOL  | FALSE   | TRUE: Interrupts processing and resets output.   |
|        | usiCom     | USINT | 1       | 1 (Fixed)  |
|        | uiDevice   | UINT  | 1       | Physical slave address   |
|        | usiChannel | USINT | 1       | Reserved   |
|        | wIndex     | WORD  | 0       | Parameter index in the object directory <sup>(Note 1)</sup>                                      |
|        | bySubIndex | BYTE  | 0       | Parameter sub-index in the object directory <sup>(Note 1)</sup>                                  |
|        | udiTimeout | UDINT | 0       | Timeout (Unit: ms)   |
| Output | xDone      | BOOL  | FALSE   | TRUE: FB processing is completed.  |
|        | xBusy      | BOOL  | FALSE   | TRUE: FB is in progress.   |

| Scope | Name          | Type                 | Default         | Description                                |
|-------|---------------|----------------------|-----------------|--|
|       | xError        | BOOL                 | FALSE           | TRUE: An error has occurred within the FB. |
|       | eError        | ETC_CO_ERROR         | ETC_CO_NO_ERROR | Error ID output                            |
|       | udiSdoAbort   | UDINT                | 0               | Abort code received from the slave device  |
|       | abyData       | ARRAY [1..4] OF BYTE | -               | Read data storage location                 |
|       | usiDataLength | USINT                | 0               | Number of read bytes                       |

(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

### 8.2.3 ETC\_CO\_SdoReadDWord (Read Double Word of Slave Parameter)

Just like ETC\_CO\_SdoRead4, this is a function block (FB) that reads the EtherCAT slave parameters. The read data is stored in DWORD (dwData), not in an array. Since byte swapping is automatically executed, read data can be directly used.

#### ■ Icon



#### ■ Parameter

| Scope | Name       | Type  | Default | Description  |
|-------|------------|-------|---------|--|
| Input | xExecute   | BOOL  | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|       | xAbort     | BOOL  | FALSE   | TRUE: Interrupts processing and resets output.   |
|       | usiCom     | USINT | 1       | 1 (Fixed)  |
|       | uiDevice   | UINT  | 0       | Physical slave address   |
|       | usiChannel | USINT | 1       | Reserved   |
|       | wIndex     | WORD  | 0       | Parameter index in the object directory <sup>(Note 1)</sup>                                      |

## 8.2 EtherCAT

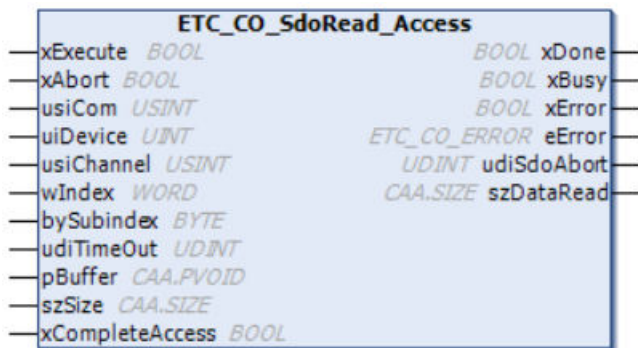
| Scope  | Name          | Type         | Default         | Description   |
|--------|---------------|--------------|-----------------|---|
|        | bySubIndex    | BYTE         | 0               | Parameter sub-index in the object directory <sup>(Note 1)</sup> |
|        | udiTimeout    | UDINT        | 0               | Timeout (Unit: ms)  |
| Output | xDone         | BOOL         | FALSE           | TRUE: FB processing is completed.                               |
|        | xBusy         | BOOL         | FALSE           | TRUE: FB is in progress.  |
|        | xError        | BOOL         | FALSE           | TRUE: An error has occurred within the FB.                      |
|        | eError        | ETC_CO_ERROR | ETC_CO_NO_ERROR | Error ID output   |
|        | udiSdoAbort   | UDINT        | 0               | Abort code received from the slave device                       |
|        | dwData        | DWORD        | 0               | Read data storage location                                      |
|        | usiDataLength | USINT        | 0               | Number of read bytes  |

(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

### 8.2.4 ETC\_CO\_SdoRead\_Access (Read Slave Parameter Index)

Just like ETC\_CO\_SdoRead, this is a function block (FB) that reads the EtherCAT slave parameters. By setting the xCompleteAccess input to TRUE and the bySubIndex input to 0, you can read complete indexes including all entries.

#### ■ Icon



#### ■ Parameter

| Scope | Name     | Type | Default | Description  |
|-------|----------|------|---------|--|
| Input | xExecute | BOOL | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|       | xAbsort  | BOOL | FALSE   | TRUE: Interrupts processing and resets output.   |



| Scope  | Name            | Type         | Default         | Description   |
|--------|-----------------|--------------|-----------------|---|
|        | usiCom          | USINT        | 1               | 1 (Fixed)   |
|        | uiDevice        | UINT         | 0               | Physical slave address  |
|        | usiChannel      | USINT        | 1               | Reserved  |
|        | wIndex          | WORD         | 0               | Parameter index in the object directory <sup>(Note 1)</sup>     |
|        | bySubIndex      | BYTE         | 0               | Parameter sub-index in the object directory <sup>(Note 1)</sup> |
|        | udiTimeout      | UDINT        | 0               | Timeout (Unit: ms)  |
|        | pBuffer         | CAA.PVOID    | 0               | Pointer to the buffer that stores read data                     |
|        | szSize          | CAA.SIZE     | 0               | Size of the buffer that stores data                             |
|        | xCompleteAccess | BOOL         | FALSE           | TRUE: Accesses all sub-indexes within the specified index.      |
| Output | xDone           | BOOL         | FALSE           | TRUE: FB processing is completed.                               |
|        | xBusy           | BOOL         | FALSE           | TRUE: FB is in progress.  |
|        | xError          | BOOL         | FALSE           | TRUE: An error has occurred within the FB.                      |
|        | eError          | ETC_CO_ERROR | ETC_CO_NO_ERROR | Error ID output   |
|        | udiSdoAbort     | UDINT        | 0               | Abort code received from the slave device                       |
|        | szDataRead      | CAA.SIZE     | 0               | Number of read bytes  |

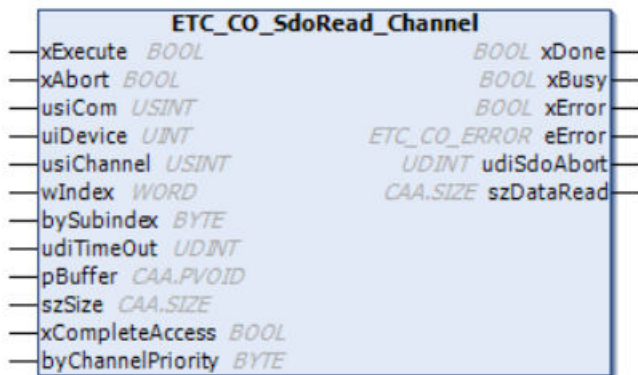
(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

### 8.2.5 ETC\_CO\_SdoRead\_Channel (Read Priority Specification of Slave Parameter)

Just like ETC\_CO\_SdoRead\_Access, this is a function block (FB) that reads the EtherCAT slave parameters. By using the byChannelPriority (BYTE) input, you can specify the channel and priority using a CoE mailbox message. Specify the channel with the first 6 bits (bit0 to bit5) and the priority with the last 2 bits (bit6 and bit7).

## 8.2 EtherCAT

### ■ Icon



### ■ Parameter

| Scope  | Name                           | Type                      | Default                      | Description  |
|--------|--------------------------------|---------------------------|------------------------------|--|
| Input  | <code>xExecute</code>          | <code>BOOL</code>         | <code>FALSE</code>           | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|        | <code>xAbort</code>            | <code>BOOL</code>         | <code>FALSE</code>           | TRUE: Interrupts processing and resets output.   |
|        | <code>usiCom</code>            | <code>USINT</code>        | <code>1</code>               | <code>1</code> (Fixed)   |
|        | <code>uiDevice</code>          | <code>UINT</code>         | <code>0</code>               | Physical slave address   |
|        | <code>usiChannel</code>        | <code>USINT</code>        | <code>1</code>               | Reserved   |
|        | <code>wIndex</code>            | <code>WORD</code>         | <code>0</code>               | Parameter index in the object directory <sup>(Note 1)</sup>                                      |
|        | <code>bySubIndex</code>        | <code>BYTE</code>         | <code>0</code>               | Parameter sub-index in the object directory <sup>(Note 1)</sup>                                  |
|        | <code>udiTimeout</code>        | <code>UDINT</code>        | <code>0</code>               | Timeout (Unit: ms)   |
|        | <code>pBuffer</code>           | <code>CAA.PVOID</code>    | <code>0</code>               | Pointer to the buffer that stores read data  |
|        | <code>szSize</code>            | <code>CAA.SIZE</code>     | <code>0</code>               | Size of the buffer that stores data  |
|        | <code>xCompleteAccess</code>   | <code>BOOL</code>         | <code>FALSE</code>           | TRUE: Accesses all sub-indexes within the specified index.                                       |
|        | <code>byChannelPriority</code> | <code>BYTE</code>         | <code>0</code>               | Specifies the channel and priority using a CoE mailbox message.                                  |
| Output | <code>xDone</code>             | <code>BOOL</code>         | <code>FALSE</code>           | TRUE: FB processing is completed.  |
|        | <code>xBusy</code>             | <code>BOOL</code>         | <code>FALSE</code>           | TRUE: FB is in progress.   |
|        | <code>xError</code>            | <code>BOOL</code>         | <code>FALSE</code>           | TRUE: An error has occurred within the FB.   |
|        | <code>eError</code>            | <code>ETC_CO_ERROR</code> | <code>ETC_CO_NO_ERROR</code> | Error ID output  |
|        | <code>udiSdoAbort</code>       | <code>UDINT</code>        | <code>0</code>               | Abort code received from the slave device  |

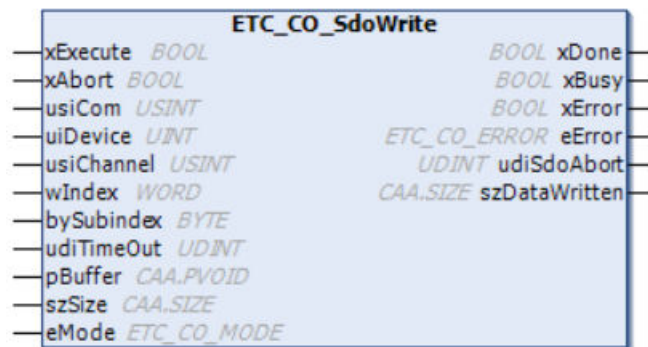
| Scope | Name       | Type     | Default | Description          |
|-------|------------|----------|---------|----------------------|
|       | szDataRead | CAA.SIZE | 0       | Number of read bytes |

(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

### 8.2.6 ETC\_CO\_SdoWrite (Write Slave Parameter)

This is a function block (FB) that writes the EtherCAT slave parameters. Unlike ETC\_CO\_SdoWrite4, this FB supports parameters longer than 4 bytes. Specify parameters to be written using the index and sub-index used for the object directory.

#### ■ Icon



#### ■ Parameter

| Scope | Name       | Type      | Default | Description  |
|-------|------------|-----------|---------|--|
| Input | xExecute   | BOOL      | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|       | xAbort     | BOOL      | FALSE   | TRUE: Interrupts processing and resets output.   |
|       | usiCom     | USINT     | 1       | 1 (Fixed)  |
|       | uiDevice   | UINT      | 0       | Physical slave address   |
|       | usiChannel | USINT     | 1       | Reserved   |
|       | wIndex     | WORD      | 0       | Parameter index in the object directory <sup>(Note 1)</sup>                                      |
|       | bySubIndex | BYTE      | 0       | Parameter sub-index in the object directory <sup>(Note 1)</sup>                                  |
|       | udiTimeout | UDINT     | 0       | Timeout (Unit: ms)   |
|       | pBuffer    | CAA.PVOID | 0       | Pointer to the buffer storing write data   |
|       | szSize     | CAA.SIZE  | 0       | Number of written bytes  |

## 8.2 EtherCAT

| Scope  | Name          | Type         | Default         | Description                                |
|--------|---------------|--------------|-----------------|--|
|        | eMode         | ETC_CO_MODE  | ETC_CO_AUTO     | Transmission mode                          |
| Output | xDone         | BOOL         | FALSE           | TRUE: FB processing is completed.          |
|        | xBusy         | BOOL         | FALSE           | TRUE: FB is in progress.                   |
|        | xError        | BOOL         | FALSE           | TRUE: An error has occurred within the FB. |
|        | eError        | ETC_CO_ERROR | ETC_CO_NO_ERROR | Error ID output                            |
|        | udiSdoAbort   | UDINT        | 0               | Abort code received from the slave device  |
|        | szDataWritten | CAA.SIZE     | 0               | Number of bytes written normally           |

(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

### ■ ETC\_CO\_ERROR (Union type)

| Member               | Type | Description                                |
|----------------------|------|--|
| ETC_CO_NO_ERROR      | WORD | No error                                   |
| ETC_CO_FIRST_ERROR   | WORD | Check udiSdoAbort for the cause of errors. |
| ETC_CO_OTHER_ERROR   | WORD | The master is not found.                   |
| ETC_CO_DATA_OVERFLOW | WORD | ETC_CO_Expedited and size exceed 4.        |
| ETC_CO_TIMEOUT       | WORD | The time limit is exceeded.                |
| ETC_CO_FIRST_MF      | WORD | Not used                                   |
| ETC_CO_LAST_ERROR    | WORD | Not used                                   |

### ■ ETC\_CO\_MODE (Union type)

| Member           | Type | Description                     |
|------------------|------|---------------------------------|
| ETC_CO_AUTO      | WORD | Mode is selected automatically. |
| ETC_CO_EXPEDITED | WORD | Expedited transfer              |
| ETC_CO_SEGMENTED | WORD | Segmented transfer              |

### 8.2.7 ETC\_CO\_SdoWrite4 (Write Four Bytes of Slave Parameter)

This is a function block (FB) that writes the EtherCAT slave parameters. Unlike ETC\_CO\_SdoWrite, this FB supports only parameters with 4 bytes or less. Specify parameters to be written using the index and sub-index used for the object directory.

## ■ Icon



## ■ Parameter

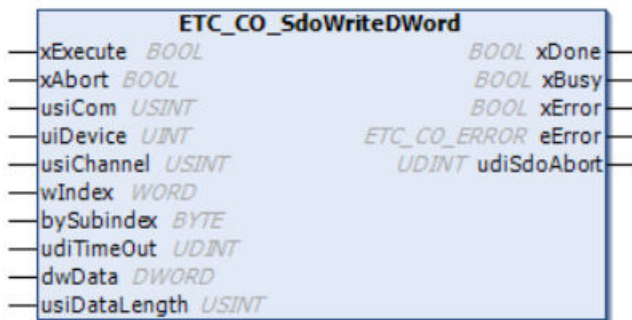
| Scope  | Name          | Type                 | Default         | Description  |
|--------|---------------|----------------------|-----------------|--|
| Input  | xExecute      | BOOL                 | FALSE           | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|        | xAbort        | BOOL                 | FALSE           | TRUE: Interrupts processing and resets output.   |
|        | usiCom        | USINT                | 1               | 1 (Fixed)  |
|        | uiDevice      | UINT                 | 0               | Physical slave address   |
|        | usiChannel    | USINT                | 1               | Reserved   |
|        | wIndex        | WORD                 | 0               | Parameter index in the object directory <sup>(Note 1)</sup>                                      |
|        | bySubIndex    | BYTE                 | 0               | Parameter sub-index in the object directory <sup>(Note 1)</sup>                                  |
|        | udiTimeout    | UDINT                | 0               | Timeout (Unit: ms)   |
|        | abyData       | ARRAY [1..4] OF BYTE | -               | Write data storage location  |
|        | usiDataLength | USINT                | 0               | Number of written bytes  |
| Output | xDone         | BOOL                 | FALSE           | TRUE: FB processing is completed.  |
|        | xBusy         | BOOL                 | FALSE           | TRUE: FB is in progress.   |
|        | xError        | BOOL                 | FALSE           | TRUE: An error has occurred within the FB.   |
|        | eError        | ETC_CO_ERROR         | ETC_CO_NO_ERROR | Error ID output  |
|        | udiSdoAbort   | UDINT                | 0               | Abort code received from the slave device  |

(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

**8.2.8 ETC\_CO\_SdoWriteDWord (Write Double Words of Slave Parameter)**

Just like ETC\_CO\_SdoWrite4, this is a function block (FB) that writes the EtherCAT slave parameters. The write data is transferred in DWORD (dwData), not in an array. Since byte swapping is automatically executed, write data can be directly used.

■ **Icon**



■ **Parameter**

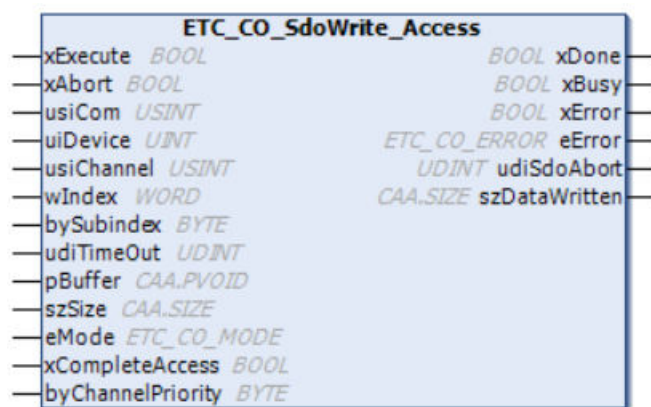
| Scope         | Name        | Type         | Default                 | Description  |
|---------------|-------------|--------------|-------------------------|--|
| Input         | xExecute    | BOOL         | FALSE                   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|               | xAbsort     | BOOL         | FALSE                   | TRUE: Interrupts processing and resets output.   |
|               | usiCom      | USINT        | 1                       | 1 (Fixed)  |
|               | uiDevice    | UINT         | 0                       | Physical slave address   |
|               | usiChannel  | USINT        | 1                       | Reserved   |
|               | wIndex      | WORD         | 0                       | Parameter index in the object directory <sup>(Note 1)</sup>                                      |
|               | bySubIndex  | BYTE         | 0                       | Parameter sub-index in the object directory <sup>(Note 1)</sup>                                  |
|               | udiTimeout  | UDINT        | 0                       | Timeout (Unit: ms)   |
|               | dwData      | DWORD        | 0                       | Write data storage location  |
| usiDataLength | USINT       | 0            | Number of written bytes |  |
| Output        | xDone       | BOOL         | FALSE                   | TRUE: FB processing is completed.  |
|               | xBusy       | BOOL         | FALSE                   | TRUE: FB is in progress.   |
|               | xError      | BOOL         | FALSE                   | TRUE: An error has occurred within the FB.   |
|               | eError      | ETC_CO_ERROR | ETC_CO_NO_ERROR         | Error ID output  |
|               | udiSdoAbort | UDINT        | 0                       | Abort code received from the slave device  |

(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

### 8.2.9 ETC\_CO\_SdoWrite\_Access (Write Slave Parameter Index)

Just like ETC\_CO\_SdoWrite, this is a function block (FB) that writes the EtherCAT slave parameters. By setting the xCompleteAccess input to TRUE and the bySubIndex input to 0, you can write complete indexes including all entries. By using the byChannelPriority (BYTE) input, you can specify the channel and priority using a CoE mailbox message. Specify the channel with the first 6 bits (bit0 to bit5) and the priority with the last 2 bits (bit6 and bit7).

#### ■ Icon



#### ■ Parameter

| Scope | Name       | Type      | Default | Description  |
|-------|------------|-----------|---------|--|
| Input | xExecute   | BOOL      | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|       | xAbort     | BOOL      | FALSE   | TRUE: Interrupts processing and resets output.   |
|       | usiCom     | USINT     | 1       | 1 (Fixed)  |
|       | uiDevice   | UINT      | 0       | Physical slave address   |
|       | usiChannel | USINT     | 1       | Reserved   |
|       | wIndex     | WORD      | 0       | Parameter index in the object directory <sup>(Note 1)</sup>                                      |
|       | bySubIndex | BYTE      | 0       | Parameter sub-index in the object directory <sup>(Note 1)</sup>                                  |
|       | udiTimeout | UDINT     | 0       | Timeout (Unit: ms)   |
|       | pBuffer    | CAA.PVOID | 0       | Pointer to the buffer storing write data   |
|       | szSize     | CAA.SIZE  | 0       | Number of written bytes  |

## 8.2 EtherCAT

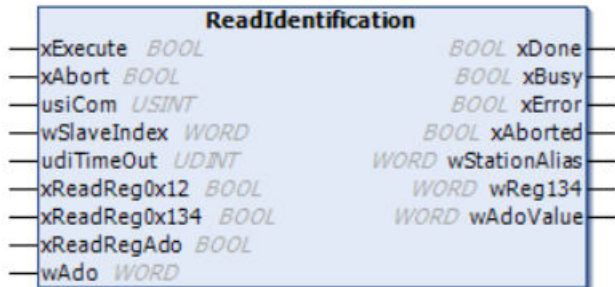
| Scope  | Name              | Type         | Default         | Description   |
|--------|-------------------|--------------|-----------------|---|
|        | eMode             | ETC_CO_MODE  | ETC_CO_AUTO     | Transmission mode   |
|        | xCompleteAccess   | BOOL         | FALSE           | TRUE: Accesses all sub-indexes within the specified index.      |
|        | byChannelPriority | BYTE         |                 | Specifies the channel and priority using a CoE mailbox message. |
| Output | xDone             | BOOL         | FALSE           | TRUE: FB processing is completed.                               |
|        | xBusy             | BOOL         | FALSE           | TRUE: FB is in progress.  |
|        | xError            | BOOL         | FALSE           | TRUE: An error has occurred within the FB.                      |
|        | eError            | ETC_CO_ERROR | ETC_CO_NO_ERROR | Error ID output   |
|        | udiSdoAbort       | UDINT        | 0               | Abort code received from the slave device                       |
|        | szDataWritten     | CAA.SIZE     | 0               | Number of bytes written normally                                |

(Note 1) The parameter content differs according to the slave. Refer to the manuals of corresponding slave devices.

### 8.2.10 ReadIdentification (Read Slave Identification Data)

Reads identification data from EtherCAT slaves.

#### ■ Icon



#### ■ Parameter

| Scope | Name        | Type  | Default | Description  |
|-------|-------------|-------|---------|--|
| Input | xExecute    | BOOL  | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|       | xAAbort     | BOOL  | FALSE   | TRUE: Interrupts processing and resets output.   |
|       | usiCom      | USINT | 1       | 1 (Fixed)  |
|       | wSlaveIndex | WORD  | 0       | Specifies EtherCAT slaves. EtherCAT slave numbers are  |

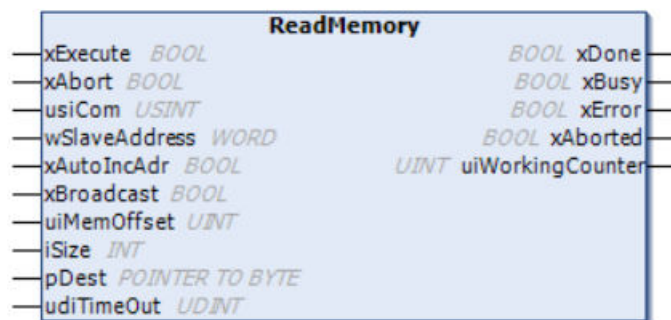


| Scope  | Name          | Type  | Default | Description  |
|--------|---------------|-------|---------|--|
|        |               |       |         | allocated in ascending order from 0 to the one closest to the master.                      |
|        | udiTimeout    | UDINT | 0       | Timeout (Unit: ms)<br>When executed with the default value 0, it will timeout immediately. |
|        | xReadReg0x12  | BOOL  | FALSE   | Register 16#12 (Station alias) read flag   |
|        | xReadReg0x134 | BOOL  | FALSE   | Register 16#134 (Explicit Device ID) read flag   |
|        | xReadRegAdo   | BOOL  | FALSE   | Register Ado read flag   |
|        | wAdo          | WORD  | 0       | Ado ID address   |
| Output | xDone         | BOOL  | FALSE   | TRUE: FB processing is completed.  |
|        | xBusy         | BOOL  | FALSE   | TRUE: FB is in progress.   |
|        | xError        | BOOL  | FALSE   | TRUE: An error has occurred within the FB.   |
|        | xAborted      | BOOL  | FALSE   | TRUE: An interruption has occurred in FB processing.                                       |
|        | wStationAlias | WORD  | 0       | Value of register 16#12  |
|        | wReg134       | WORD  | 0       | Value of register 16#0x134   |
|        | wAdoValue     | WORD  | 0       | Value of Ado ID  |

### 8.2.11 ReadMemory (Read Slave Memory)

This is a function block (FB) that reads the EtherCAT slave memory.

#### ■ Icon



#### ■ Parameter

| Scope | Name     | Type | Default | Description                                |
|-------|----------|------|---------|--|
| Input | xExecute | BOOL | FALSE   | TRUE: Starts execution at the rising edge. |

## 8.2 EtherCAT

| Scope  | Name             | Type            | Default | Description   |
|--------|------------------|-----------------|---------|---|
|        |                  |                 |         | FALSE: Resets output after execution is completed.  |
|        | xAbort           | BOOL            | FALSE   | TRUE: Interrupts processing and resets output.  |
|        | usiCom           | USINT           | 1       | Master device index (1 onwards)   |
|        | wSlaveAddress    | WORD            | 0       | Either the automatic incremental address or the device's physical address   |
|        | xAutoIncAdr      | BOOL            | FALSE   | TRUE: Uses the automatic incremental address.<br>When set to TRUE, specify the automatic incremental address for wSlaveAddress. |
|        | xBroadcast       | BOOL            | FALSE   | TRUE: Uses the broadcast read.<br>If set to TRUE, wSlaveAddress and bAutoIncAdr are not used.                                   |
|        | uiMemOffset      | UINT            | 0       | Offset of the memory  |
|        | iSize            | INT             | 0       | Number of read bytes  |
|        | pDest            | POINTER TO BYTE | 0       | Pointer to the buffer that stores read data   |
|        | udiTimeOut       | UDINT           | 0       | Timeout (Unit: ms)<br>When executed with the default value 0, it will timeout immediately.                                      |
| Output | xDone            | BOOL            | FALSE   | TRUE: FB processing is completed.   |
|        | xBusy            | BOOL            | FALSE   | TRUE: FB is in progress.  |
|        | xError           | BOOL            | FALSE   | TRUE: An error has occurred within the FB.  |
|        | xAborted         | BOOL            | FALSE   | TRUE: An interruption has occurred in FB processing.  |
|        | uiWorkingCounter | UINT            | 0       | Working counter of received commands  |

### 8.2.12 ReadNbrSlaves (Read the Number of Connected Slaves)

Reads the number of slaves currently connected.

#### ■ Icon



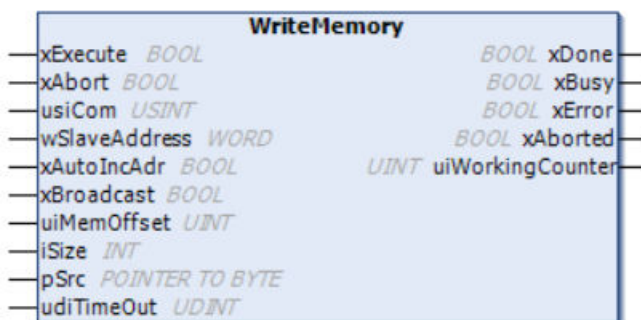
### Parameter

| Scope  | Name          | Type  | Default | Description  |
|--------|---------------|-------|---------|--|
| Input  | xExecute      | BOOL  | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |
|        | xAbort        | BOOL  | FALSE   | TRUE: Interrupts processing and resets output.   |
|        | usiCom        | USINT | 1       | 1 (Fixed)  |
|        | udiTimeOut    | UDINT | 0       | Timeout (Unit: ms)<br>When executed with the default value 0, it will timeout immediately.       |
| Output | xDone         | BOOL  | FALSE   | TRUE: FB processing is completed.  |
|        | xBusy         | BOOL  | FALSE   | TRUE: FB is in progress.   |
|        | xError        | BOOL  | FALSE   | TRUE: An error has occurred within the FB.   |
|        | xAborted      | BOOL  | FALSE   | TRUE: An interruption has occurred in FB processing.   |
|        | wNumberSlaves | WORD  | 0       | Number of slaves connected   |

### 8.2.13 WriteMemory (Write Slave Memory)

This is a function block (FB) that writes the EtherCAT slave memory. Accesses "ESC address space".

#### Icon



#### Parameter

| Scope | Name     | Type | Default | Description  |
|-------|----------|------|---------|--|
| Input | xExecute | BOOL | FALSE   | TRUE: Starts execution at the rising edge.<br>FALSE: Resets output after execution is completed. |

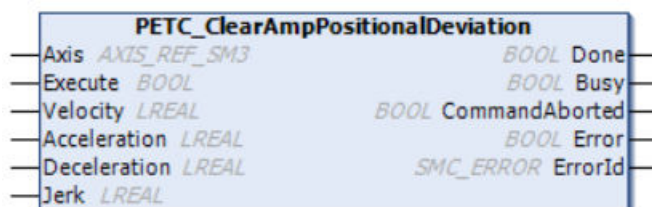
## 8.2 EtherCAT

| Scope  | Name             | Type            | Default | Description   |
|--------|------------------|-----------------|---------|---|
|        | xAbort           | BOOL            | FALSE   | TRUE: Interrupts processing and resets output.  |
|        | usiCom           | USINT           | 1       | 1 (Fixed)   |
|        | wSlaveAddress    | WORD            | 0       | When set to the physical slave address or to bAutoIncrementAddress, specify the automatic incremental address.                  |
|        | xAutoIncAdr      | BOOL            | FALSE   | TRUE: Uses the automatic incremental address.<br>When set to TRUE, specify the automatic incremental address for wSlaveAddress. |
|        | xBroadcast       | BOOL            | FALSE   | TRUE: Uses the broadcast read.<br>If set to TRUE, wSlaveAddress and bAutoIncAdr are not used.                                   |
|        | uiMemOffset      | UINT            | 0       | Offset of the memory  |
|        | iSize            | INT             | 0       | Number of bytes to be written   |
|        | pSrc             | POINTER TO BYTE | 0       | Address of the data buffer to be written  |
|        | udiTimeOut       | UDINT           | 0       | Timeout (Unit: ms)<br>When executed with the default value 0, it will timeout immediately.                                      |
| Output | xDone            | BOOL            | FALSE   | TRUE: FB processing is completed.   |
|        | xBusy            | BOOL            | FALSE   | TRUE: FB is in progress.  |
|        | xError           | BOOL            | FALSE   | TRUE: An error has occurred within the FB.  |
|        | xAborted         | BOOL            | FALSE   | TRUE: An interruption has occurred in FB processing.  |
|        | uiWorkingCounter | UINT            | 0       | Working counter of received commands  |

### 8.2.14 PETC\_ClearAmpPositionalDeviation (Clear Amplifier Deviation Counter)

This is a function block (FB) that clears the deviation counter of the AMP. It deletes the position deviation data in the deviation counter of the AMP.

#### ■ Icon



### ■ Parameter

| Scope          | Name                             | Type                        | Default | Description                                       |
|----------------|----------------------------------|-----------------------------|---------|---|
| Input / output | Axis                             | AXIS_REF RTE<br>X_Panasonic | -       | Specifies the axis.                               |
| Input          | Execute                          | BOOL                        | FALSE   | Starts execution at the rising edge.              |
|                | Velocity <sup>(Note 1)</sup>     | LREAL                       |         | Information required to execute MC_MoveAbsolute   |
|                | Acceleration <sup>(Note 1)</sup> | LREAL                       |         | Information required to execute MC_MoveAbsolute   |
|                | Deceleration <sup>(Note 1)</sup> | LREAL                       |         | Information required to execute MC_MoveAbsolute   |
|                | Jerk <sup>(Note 1)</sup>         | LREAL                       |         | Information required to execute MC_MoveAbsolute   |
| Output         | Done                             | BOOL                        | FALSE   | TRUE: Clearing is completed.                      |
|                | Busy                             | BOOL                        | FALSE   | TRUE: Execution of the FB is not completed.       |
|                | CommandAborted                   | BOOL                        | FALSE   | TRUE: An interruption from other FB has occurred. |
|                | Error                            | BOOL                        | FALSE   | TRUE: An error has occurred within the FB.        |
|                | ErrorID                          | SMC_ERROR                   | 0       | Error ID output                                   |

(Note 1) This function block internally substitutes the command position with an actual position to call MC\_MoveAbsolute and, therefore, requires parameters including Velocity, Acceleration, Deceleration, and Jerk.

The PETC\_ClearAmpPositionalDeviation function block outputs the following errors.

| Error                          | Description   |
|--------------------------------|---|
| SMC_WRONG_CONTROLLER_MODE      | Executed in a mode other than the position control mode.<br>Change to SMC_position using SMC_SetControllerMode.             |
| SMC_AXIS_NOT_READY_FOR_MOTION  | The axis is in a state where PMC_ClearAmpMultiTurnData cannot be executed. It can be executed only at the standstill state. |
| SMC_REGULATOR_OR_START_NOT_SET | The axis is in a servo OFF state.   |
| SMC_PP_WRONG_AXIS_TYPE         | The axis is a virtual axis.   |

#### 8.2.15 IoDrvEtherCAT.GetStatistics (Get EtherCAT Frame Statistics Information)

Gets the EtherCAT frame statistics information.

| Scope  | Name              | Type  | Description                    |
|--------|-------------------|-------|--------------------------------|
| Output | udiSendFrameCount | UDINT | Number of EtherCAT send frames |

## 8.2 EtherCAT

---

| Scope | Name               | Type  | Description                               |
|-------|--------------------|-------|---|
|       | udiFramesPerSecond | UDINT | Number of EtherCAT send frames per second |
|       | udiLostFrameCount  | UDINT | Number of missing EtherCAT frames         |
|       | udiTxErrorCount    | UDINT | Number of send errors                     |
|       | udiRxErrorCount    | UDINT | Number of receive errors                  |

# 9 Motion Control Function Blocks (Auxiliary Function)

---

|  |      |
|--|------|
| 9.1 Motion Auxiliary Function (Monitoring).....                                  | 9-2  |
| 9.1.1 MC_ReadActualPosition (Read Current Position) .....                        | 9-2  |
| 9.1.2 MC_ReadActualVelocity (Read Current Velocity) .....                        | 9-2  |
| 9.1.3 PMC_ReadActualTorque (Read Current Torque) .....                           | 9-3  |
| 9.1.4 MC_ReadActualTorque (Read Current Torque).....                             | 9-4  |
| 9.1.5 MC_ReadAxisError (Read Axis Error) .....                                   | 9-4  |
| 9.1.6 MC_ReadStatus (Read Status) .....  | 9-5  |
| 9.1.7 SMC_InPosition (In-position Judgment) .....                                | 9-7  |
| 9.1.8 SMC_ReadFBError (Read Oldest Error) .....                                  | 9-8  |
| 9.1.9 SMC_ClearFBError (Clear Oldest Error) .....                                | 9-9  |
| 9.1.10 SMC_CheckAxisCommunication (Check Axis Communication<br>Status).....      | 9-10 |
| 9.1.11 SMC_CheckLimits (Check Exceeding Limits).....                             | 9-11 |
| 9.1.12 SMC_GetMaxSetAccDec (Measure Maximum Acceleration /<br>Deceleration)..... | 9-12 |
| 9.1.13 SMC_GetMaxSetVelocity (Measure Maximum Velocity) .....                    | 9-12 |
| 9.1.14 SMC_GetTrackingError (Measure Tracking Error).....                        | 9-13 |
| 9.1.15 SMC_MeasureDistance (Measure Turnaround Travel Distance) ..               | 9-14 |
| 9.1.16 SMC_ReadSetPosition (Read Axis Set Position) .....                        | 9-15 |
| 9.2 Motion Auxiliary Function (Change / Reset) .....                             | 9-16 |
| 9.2.1 MC_Reset (Axis Error Reset) .....  | 9-16 |
| 9.2.2 SMC3_ReinitDrive (Reinitialize Axis).....                                  | 9-16 |
| 9.2.3 MC_SetPosition (Change Current Position) .....                             | 9-17 |
| 9.3 Motion Auxiliary Function (Other Functions).....                             | 9-18 |
| 9.3.1 PMC_ReadLatchPosition (Amplifier Latch Monitor).....                       | 9-18 |
| 9.3.2 PMC_StopLatchPosition (Stop Amplifier Latch) .....                         | 9-20 |
| 9.3.3 MC_TouchProbe (Enable AMP Latch Monitoring) .....                          | 9-23 |
| 9.3.4 MC_AbortTrigger (Disable AMP Latch Monitoring).....                        | 9-25 |
| 9.3.5 MC_DigitalCamSwitch (Enable Digital Cam Switch) .....                      | 9-26 |
| 9.3.6 SMC_BacklashCompensation (Compensate Backlash).....                        | 9-29 |

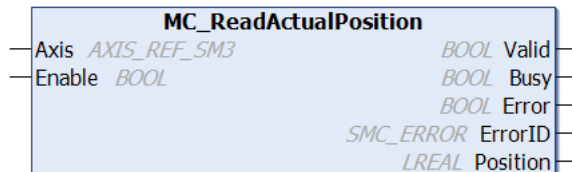
## 9.1 Motion Auxiliary Function (Monitoring)

### 9.1 Motion Auxiliary Function (Monitoring)

#### 9.1.1 MC\_ReadActualPosition (Read Current Position)

This is a function block (FB) that reads the actual position data of the axis.

##### ■ Icon



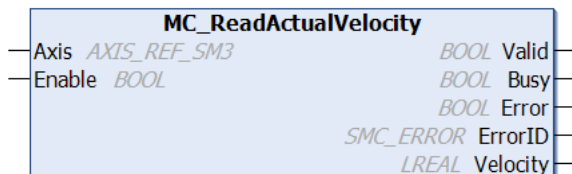
##### ■ Parameter

| Scope          | Name     | Type         | Initial | Description  |
|----------------|----------|--------------|---------|--|
| Input / output | Axis     | AXIS_REF_SM3 | -       | Specifies the axis.                                    |
| Input          | Enable   | BOOL         | FALSE   | Reads the actual position while Enable is set to TRUE. |
| Output         | Valid    | BOOL         | FALSE   | TRUE: Valid output                                     |
|                | Busy     | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.            |
|                | Error    | BOOL         | FALSE   | TRUE: An error has occurred within the FB.             |
|                | ErrorID  | SMC_ERROR    | 0       | An error ID is output.                                 |
|                | Position | LREAL        | 0       | Actual position (u) that is read out                   |

#### 9.1.2 MC\_ReadActualVelocity (Read Current Velocity)

This is a function block (FB) that reads the actual velocity of the axis.

##### ■ Icon





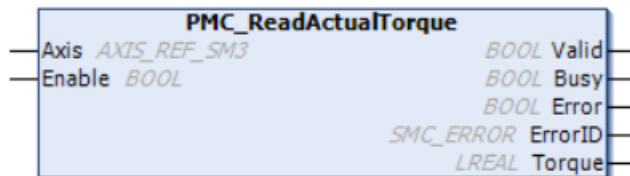
### ■ Parameter

| Scope          | Name     | Type         | Initial | Description  |
|----------------|----------|--------------|---------|--|
| Input / output | Axis     | AXIS_REF_SM3 | -       | Specifies the axis.                                    |
| Input          | Enable   | BOOL         | FALSE   | Reads the actual velocity while Enable is set to TRUE. |
| Output         | Valid    | BOOL         | FALSE   | TRUE: Valid output                                     |
|                | Busy     | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.            |
|                | Error    | BOOL         | FALSE   | TRUE: An error has occurred within the FB.             |
|                | ErrorID  | SMC_ERROR    | 0       | An error ID is output.                                 |
|                | Velocity | LREAL        | 0       | Current actual velocity (u/s) that is read out         |

### 9.1.3 PMC\_ReadActualTorque (Read Current Torque)

This is a function block (FB) that reads the actual torque value of the axis.

### ■ Icon



### ■ Parameter

| Scope          | Name    | Type         | Initial | Description  |
|----------------|---------|--------------|---------|--|
| Input / output | Axis    | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Enable  | BOOL         | FALSE   | Reads the actual torque value while Enable is set to TRUE. |
| Output         | Valid   | BOOL         | FALSE   | TRUE: Valid output   |
|                | Busy    | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                |
|                | Error   | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                 |
|                | ErrorID | SMC_ERROR    | 0       | An error ID is output.                                     |
|                | Torque  | LREAL        | 0       | Current actual torque (%) that is read out                 |

## 9.1 Motion Auxiliary Function (Monitoring)

### 9.1.4 MC\_ReadActualTorque (Read Current Torque)

This is a function block (FB) that reads the current torque value of the axis.

#### ■ Icon



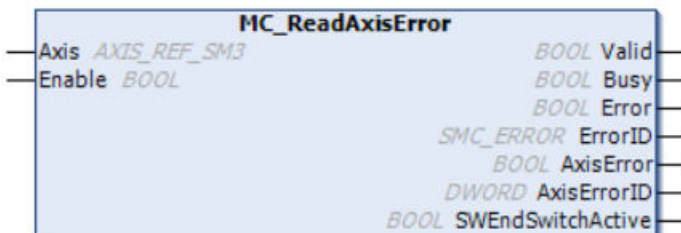
#### ■ Parameter

| Scope          | Name    | Type         | Default | Description  |
|----------------|---------|--------------|---------|--|
| Input / output | Axis    | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Enable  | BOOL         | FALSE   | Reads the actual torque value while Enable is set to TRUE. |
| Output         | Valid   | BOOL         | FALSE   | TRUE: Valid output   |
|                | Busy    | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.                |
|                | Error   | BOOL         | FALSE   | TRUE: An error has occurred within the FB.                 |
|                | ErrorID | SMC_ERROR    | 0       | Error ID output  |
|                | Torque  | LREAL        | 0       | Current actual torque that is read out (N·m, N)            |

### 9.1.5 MC\_ReadAxisError (Read Axis Error)

This is a function block that gets general axis errors not related to function blocks.

#### ■ Icon



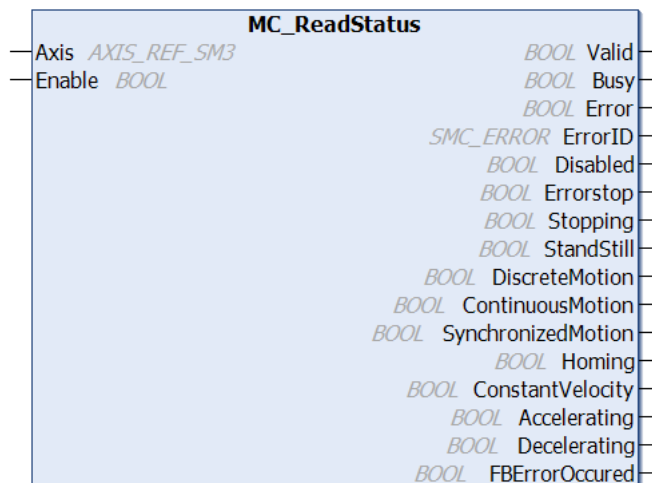
### ■ Parameter

| Scope          | Name              | Type         | Default | Description  |
|----------------|-------------------|--------------|---------|--|
| Input / output | Axis              | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Enable            | BOOL         | FALSE   | Reads the state while Enable is set to TRUE.   |
| Output         | Valid             | BOOL         | FALSE   | TRUE: Valid output   |
|                | Busy              | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.  |
|                | Error             | BOOL         | FALSE   | TRUE: An error has occurred within the FB.   |
|                | ErrorID           | SMC_ERROR    | 0       | Error ID output  |
|                | AxisError         | BOOL         | FALSE   | TRUE: An axis error has occurred.  |
|                | AxisErrorID       | DWORD        | 0       | Axis error ID<br>Servo amplifier alarms can be acquired from AxisErrorID. If an Err27.4 error occurs when connected to the MINAS, 27 (16#1B) is set and AxisErrorID becomes 16#0000FF1B. |
|                | SWEndSwitchActive | BOOL         | FALSE   | TRUE: The software limit has been exceeded.  |

### 9.1.6 MC\_ReadStatus (Read Status)

This is a function block (FB) that reads the status information of the axis. It reads detailed information about the axis state.

### ■ Icon



## 9.1 Motion Auxiliary Function (Monitoring)

### ■ Parameter

| Scope          | Name               | Type         | Initial | Description   |
|----------------|--------------------|--------------|---------|---|
| Input / output | Axis               | AXIS_REF_SM3 | -       | Specifies the axis.                                   |
| Input          | Enable             | BOOL         | FALSE   | Reads the status information while the input is TRUE. |
| Output         | Valid              | BOOL         | FALSE   | TRUE: Valid output                                    |
|                | Busy               | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.           |
|                | Error              | BOOL         | FALSE   | TRUE: An error has occurred within the FB.            |
|                | ErrorID            | SMC_ERROR    | 0       | An error ID is output.                                |
|                | Disabled           | BOOL         | FALSE   | TRUE: The axis is in the Disabled state.              |
|                | ErrorStop          | BOOL         | FALSE   | TRUE: The axis is in the ErrorStop state.             |
|                | Stopping           | BOOL         | FALSE   | TRUE: The axis is in the Stopping state.              |
|                | StandStill         | BOOL         | FALSE   | TRUE: The axis is in the StandStill state.            |
|                | DiscreteMotion     | BOOL         | FALSE   | TRUE: The axis is in the DiscreteMotion state.        |
|                | ContinuousMotion   | BOOL         | FALSE   | TRUE: The axis is in the ContinuousMotion state.      |
|                | SynchronizedMotion | BOOL         | FALSE   | TRUE: The axis is in the SynchronizedMotion state.    |
|                | Homing             | BOOL         | FALSE   | TRUE: The axis is in the Homing state.                |
|                | ConstantVelocity   | BOOL         | FALSE   | TRUE: The axis is moving at a constant velocity.      |
|                | Accelerating       | BOOL         | FALSE   | TRUE: The axis is moving in acceleration.             |
|                | Decelerating       | BOOL         | FALSE   | TRUE: The axis is moving in deceleration.             |
|                | FBErrorOccured     | BOOL         | FALSE   | TRUE: An FB error has occurred.                       |

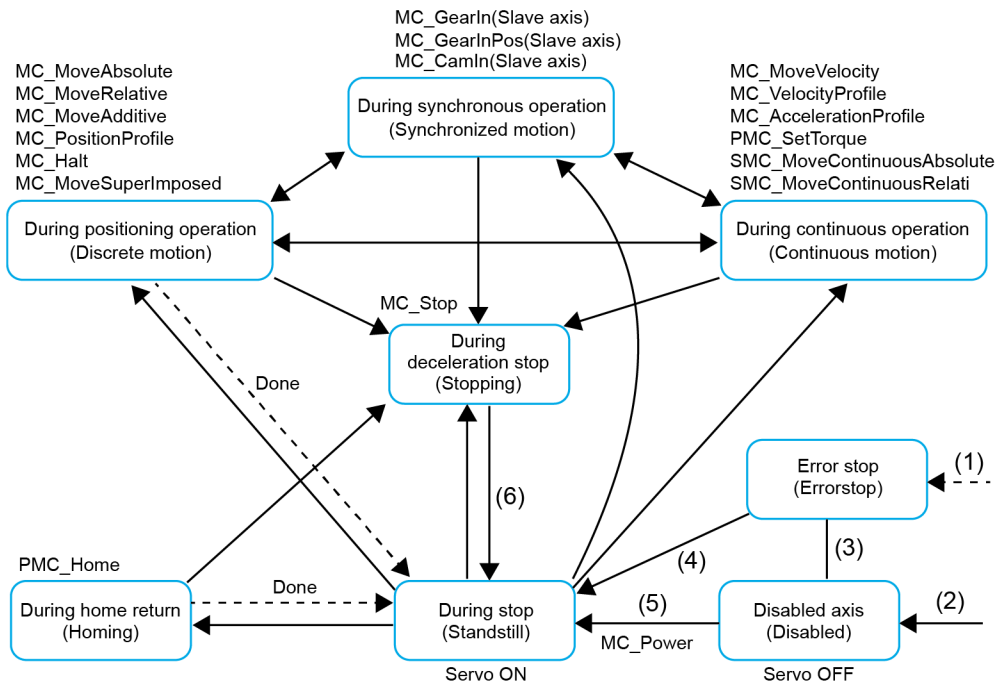
### ■ Axis state

The following section describes state transition diagram of the axis when the motion function blocks are executed.

#### State transition diagram

- The blue frame indicates the state.
- When the function block indicated above the state is executed, the state transitions to the direction indicated by the solid-line arrow.
- When the execution is completed or when an error occurs, the state transitions to the state indicated at the tip of the broken-line arrow.
- The terms in parentheses are defined in PLCopen.

## 9.1 Motion Auxiliary Function (Monitoring)



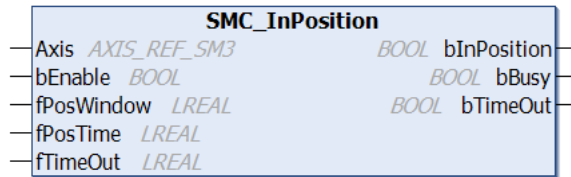
| Num<br>ber | Transition conditions  |
|------------|--|
| (1)        | Regardless of the state, when an error occurs in the axis  |
| (2)        | Regardless of the state, when Enable of MC_Power is TRUE, bRegulator is FALSE, and there is no error in the axis |
| (3)        | When Status of MC_Reset and Status of MC_Power are FALSE   |
| (4)        | When Enable of MC_Reset and Enable of MC_Power are TRUE, bRegulator is TRUE, and Status is TRUE                  |
| (5)        | When Enable of MC_Power is TRUE, bRegulator is TRUE, and Status is TRUE  |
| (6)        | When Done of MC_Stop is TRUE and Execute of MC_Stop is FALSE   |

### 9.1.7 SMC\_InPosition (In-position Judgment)

This is a function block (FB) that compares the actual position of the AMP with the command value and judges whether the position is within the specified range. The maximum difference between the actual position of the AMP and the command value as well as the dwell time are specified to judge (in-position judgment) whether the specified values are satisfied.

## 9.1 Motion Auxiliary Function (Monitoring)

### ■ Icon



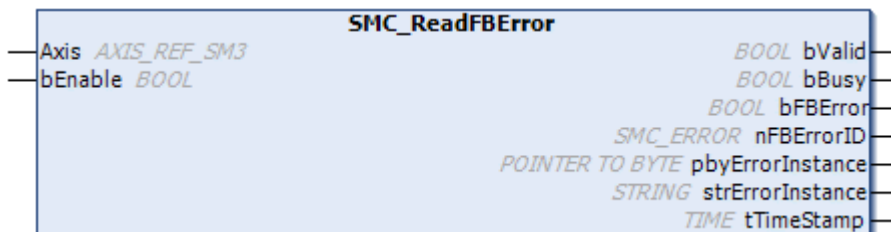
### ■ Parameter

| Scope          | Name        | Type         | Initial | Description  |
|----------------|-------------|--------------|---------|--|
| Input / output | Axis        | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | bEnable     | BOOL         | FALSE   | TRUE: Executes the FB.   |
|                | fPosWindow  | LREAL        | 0       | The maximum difference between the actual position and the command value to judge whether the target position has been reached.  |
|                | fPosTime    | LREAL        | 0       | The dwell time (s) to judge whether the axis has reached the position  |
|                | fTimeOut    | LREAL        | 0       | The time (s) from when the FB is enabled to when judgment is made that timeout has occurred<br>When the value is "0", the timeout judgment is not made yet.                            |
| Output         | bInPosition | BOOL         | FALSE   | TRUE: The target position is reached.<br>While the difference between the actual position and the command value is within the time specified in fPosTime, it is within the fPosWindow. |
|                | bBusy       | BOOL         | FALSE   | TRUE: The FB is in operation.  |
|                | bTimeOut    | BOOL         | FALSE   | TRUE: Timeout has occurred.  |

### 9.1.8 SMC\_ReadFBError (Read Oldest Error)

This is a function block (FB) that reads the oldest function block error information.

### ■ Icon



### ■ Parameter

| Scope          | Name             | Type            | Initial  | Description                                      |
|----------------|------------------|-----------------|----------|--|
| Input / output | Axis             | AXIS_REF_SM3    | -        | Specifies the axis.                              |
| Input          | bEnable          | BOOL            | FALSE    | TRUE: The FB can be executed.                    |
| Output         | bValid           | BOOL            | FALSE    | TRUE: Error information is acquired.             |
|                | bBusy            | BOOL            | FALSE    | TRUE: The FB is in operation.                    |
|                | bFBError         | BOOL            | FALSE    | TRUE: An error has occurred within the FB.       |
|                | nFBErrorID       | SMC_ERROR       | 0        | An error ID is output.                           |
|                | pbyErrorInstance | POINTER TO BYTE | 0        | FB instance of the error acquisition source      |
|                | strErrorInstance | STRING          | "        | FB instance name of the error acquisition source |
|                | tTimeStamp       | TIME            | TIME#0ms | Time stamp of the error information              |

### Note

- The error information is cleared when SMC\_ClearFBError is executed. When an error occurs again, SMC\_ReadFBError reads the error.

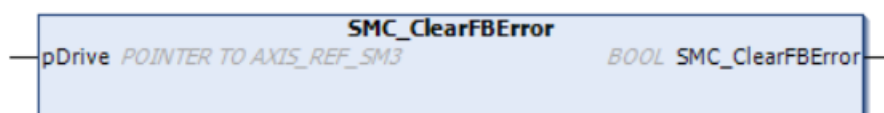
### REFERENCE

[9.1.9 SMC\\_ClearFBError \(Clear Oldest Error\)](#)

### 9.1.9 SMC\_ClearFBError (Clear Oldest Error)

This function clears the oldest FB error information.

### ■ Icon



## 9.1 Motion Auxiliary Function (Monitoring)

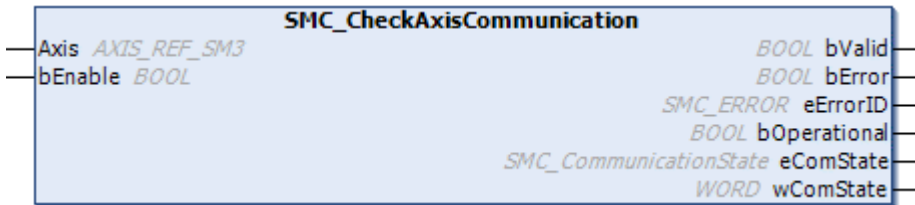
### ■ Parameter

| Type   | Parameter name   | Type                       | Default | Description  |
|--------|------------------|----------------------------|---------|--|
| Input  | pDrive           | POINTER TO<br>AXIS_REF_SM3 | -       | Specifies the axis.  |
| Return | SMC_ClearFBError | BOOL                       |         | This function always returns FALSE even for normal completion. |

### 9.1.10 SMC\_CheckAxisCommunication (Check Axis Communication Status)

This is a function block (FB) that checks the communication state of the axis.

### ■ Icon



### ■ Parameter

| Scope          | Name         | Type                   | Initial                  | Description                                     |
|----------------|--------------|------------------------|--------------------------|---|
| Input / output | Axis         | AXIS_REF_SM3           | -                        | Specifies the axis.                             |
| Input          | bEnable      | BOOL                   | FALSE                    | TRUE: The FB can be executed.                   |
| Output         | bValid       | BOOL                   | FALSE                    | TRUE: The output value is valid.                |
|                | bError       | BOOL                   | FALSE                    | TRUE: An error has occurred within the FB.      |
|                | eErrorID     | SMC_ERROR              | 0                        | An error ID is output.                          |
|                | bOperational | BOOL                   | FALSE                    | TRUE: Communication state is operational. (100) |
|                | eComState    | SMC_CommunicationState | SMC_COMSTATE_NOT_STARTED | Communication state                             |
|                | wComState    | WORD                   | 0                        | Internal value of the communication state       |

### ■ SMC\_CommunicationState (Enumeration type)

| Name                                 | Value | Description                 |
|--------------------------------------|-------|-----------------------------|
| SMC_COMSTATE_NOT_STARTED             | 0     | Stop                        |
| SMC_COMSTATE_VARIABLE_INITIALIZATION | 1     | Initialization of variables |



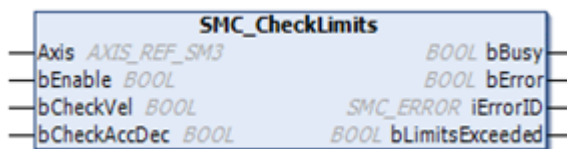
## 9.1 Motion Auxiliary Function (Monitoring)

| Name                                 | Value | Description                                   |
|--------------------------------------|-------|---|
| SMC_COMSTATE_BASE_COM_INITIALIZATION | 2     | Initialization of base communication settings |
| SMC_COMSTATE_DRIVE_INITIALIZATION    | 3     | Initialization of drive settings              |
| SMC_COMSTATE_DRIVE_WAITING_FOR_SYNC  | 4     | Waiting for drive synchronization             |
| SMC_COMSTATE_INITIALIZATION_DONE     | 5     | Initialization completed                      |
| SMC_COMSTATE_OPERATIONAL             | 6     | Operational                                   |
| SMC_COMSTATE_REINITIALIZATION        | 7     | Re-initialization                             |
| SMC_COMSTATE_ERROR                   | 8     | Error   |
| SMC_COMSTATE_UNKNOWN                 | 9     | Unknown                                       |

### 9.1.11 SMC\_CheckLimits (Check Exceeding Limits)

This is a function block (FB) that checks whether the velocity, acceleration, or deceleration is in excess of the dynamic limit set value of the device.

#### ■ Icon



#### ■ Parameter

| Scope          | Name            | Type         | Initial | Description  |
|----------------|-----------------|--------------|---------|--|
| Input / output | Axis            | AXIS_REF_SM3 | -       | Specifies the axis.                                      |
| Input          | bEnable         | BOOL         | FALSE   | TRUE: The FB can be executed.                            |
|                | bCheckVel       | BOOL         | TRUE    | TRUE: Checks the velocity setting.                       |
|                | bCheckAccDec    | BOOL         | FALSE   | TRUE: Checks the acceleration and deceleration settings. |
| Output         | bBusy           | BOOL         | FALSE   | TRUE: The FB is in operation.                            |
|                | bError          | BOOL         | FALSE   | TRUE: An error has occurred within the FB.               |
|                | iErrorID        | SMC_ERROR    | 0       | An error ID is output.                                   |
|                | bLimitsExceeded | BOOL         | FALSE   | TRUE: Limits are exceeded.                               |

## 9.1 Motion Auxiliary Function (Monitoring)

### Info.

- Reference manual  
*GM1 Controller RTEX User's Manual (Operation Edition)*  
*GM1 Controller EtherCAT User's Manual (Operation Edition)*

### 9.1.12 SMC\_GetMaxSetAccDec (Measure Maximum Acceleration / Deceleration)

This is a function block (FB) that measures the maximum value of the axis acceleration/ deceleration command.

#### ■ Icon



#### ■ Parameter

| Scope          | Name             | Type         | Initial | Description                                   |
|----------------|------------------|--------------|---------|---|
| Input / output | Axis             | AXIS_REF_SM3 | -       | Specifies the axis.                           |
| Input          | bEnable          | BOOL         | FALSE   | TRUE: The FB can be executed.                 |
|                | dwTimeStamp      | DWORD        | 0       | Time stamp                                    |
| Output         | bValid           | BOOL         | FALSE   | TRUE: The output value is valid.              |
|                | bBusy            | BOOL         | FALSE   | TRUE: The FB is in operation.                 |
|                | fMaxAcceleration | LREAL        | 0       | Maximum acceleration (u/s <sup>2</sup> ).     |
|                | dwTimeAtMax      | DWORD        | 0       | dwTimeStamp value at the maximum acceleration |

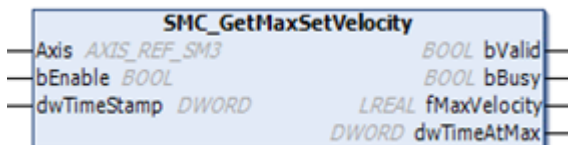
### Note

- It is possible to check when the maximum acceleration or deceleration has occurred by entering a call counter value in the input variable "dwTimeStamp".

### 9.1.13 SMC\_GetMaxSetVelocity (Measure Maximum Velocity)

This is a function block (FB) that measures the maximum value of the axis velocity command.

### ■ Icon



### ■ Parameter

| Scope          | Name         | Type         | Initial | Description                                   |
|----------------|--------------|--------------|---------|---|
| Input / output | Axis         | AXIS_REF_SM3 | -       | Specifies the axis.                           |
| Input          | bEnable      | BOOL         | FALSE   | TRUE: The FB can be executed.                 |
|                | dwTimeStamp  | DWORD        | 0       | Time stamp                                    |
| Output         | bValid       | BOOL         | FALSE   | TRUE: The output value is valid.              |
|                | bBusy        | BOOL         | FALSE   | TRUE: The FB is in operation.                 |
|                | fMaxVelocity | LREAL        | 0       | Maximum velocity (u/s).                       |
|                | dwTimeAtMax  | DWORD        | 0       | dwTimeStamp value at the maximum acceleration |

### ■ Note

- It is possible to check when the maximum velocity has occurred by entering a call counter value in the input variable "dwTimeStamp".

### 9.1.14 SMC\_GetTrackingError (Measure Tracking Error)

This is a function block (FB) that measures the tracking error of the actual position for the axis command position.

### ■ Icon



### ■ Parameter

| Scope          | Name             | Type         | Initial | Description                   |
|----------------|------------------|--------------|---------|-------------------------------|
| Input / output | Axis             | AXIS_REF_SM3 | -       | Specifies the axis.           |
| Input          | bEnable          | BOOL         | FALSE   | TRUE: The FB can be executed. |
|                | byDeadTimeCycles | BYTE         | 2       | Number of dead time cycles    |

## 9.1 Motion Auxiliary Function (Monitoring)

| Scope  | Name              | Type  | Initial | Description   |
|--------|-------------------|-------|---------|---|
|        |                   |       |         | Compares the command position and actual position between the specified cycles. |
|        | dwTimeStamp       | DWORD | 0       | Time stamp  |
| Output | bValid            | BOOL  | FALSE   | TRUE: The output value is valid.  |
|        | bBusy             | BOOL  | FALSE   | TRUE: The FB is in operation.   |
|        | fActTrackingError | LREAL | 0       | Actual tracking error   |
|        | fMaxTrackingError | LREAL | 0       | Maximum tracking error while the function block is being executed               |
|        | dwTimeAtMax       | DWORD | 0       | dwTimeStamp value when the maximum tracking error is detected                   |

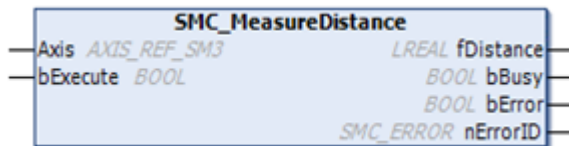
### Note

- It is possible to check when the maximum tracking error has occurred by entering a call counter value in the input variable "dwTimeStamp".

### 9.1.15 SMC\_MeasureDistance (Measure Turnaround Travel Distance)

This is a function block (FB) that measures the travel distance. For the modulo axis, the cover distance can be measured considering the laps.

#### Icon



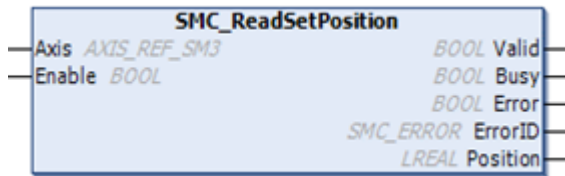
#### Parameter

| Scope          | Name      | Type         | Initial | Description  |
|----------------|-----------|--------------|---------|--|
| Input / output | Axis      | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | bExecute  | BOOL         | FALSE   | TRUE: Starts measurement at the rising edge.<br>FALSE: Ends measurement. |
| Output         | fDistance | LREAL        | 0       | Distance traveled from the start of measurement                          |
|                | bBusy     | BOOL         | FALSE   | TRUE: The FB is in operation.  |
|                | bError    | BOOL         | 0       | TRUE: An error has occurred within the FB.                               |
|                | nErrorID  | SMC_ERROR    | 0       | An error ID is output.   |

### 9.1.16 SMC\_ReadSetPosition (Read Axis Set Position)

This is a function block (FB) that acquires the command position of the axis.

#### ■ Icon



#### ■ Parameter

| Scope          | Name     | Type         | Initial | Description                                |
|----------------|----------|--------------|---------|--|
| Input / output | Axis     | AXIS_REF_SM3 | -       | Specifies the axis.                        |
| Input          | Enable   | BOOL         | FALSE   | TRUE: Executes the FB.                     |
| Output         | Valid    | BOOL         | FALSE   | TRUE: The output value is valid.           |
|                | Busy     | BOOL         | FALSE   | TRUE: The FB is in operation.              |
|                | Error    | BOOL         | FALSE   | TRUE: An error has occurred within the FB. |
|                | ErrorID  | SMC_ERROR    | 0       | An error ID is output.                     |
|                | Position | LREAL        | 0       | Axis position                              |

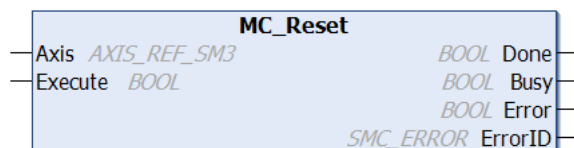
## 9.2 Motion Auxiliary Function (Change / Reset)

### 9.2 Motion Auxiliary Function (Change / Reset)

#### 9.2.1 MC\_Reset (Axis Error Reset)

This is a function block (FB) that resets the state transition error of the axis. It reset the axis error and transitions the state from the ErrorStop state to the StandStill state.

##### ■ Icon



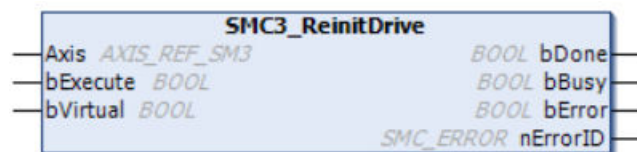
##### ■ Parameter

| Scope          | Name    | Type         | Initial | Description                                 |
|----------------|---------|--------------|---------|---|
| Input / output | Axis    | AXIS_REF_SM3 | -       | Specifies the axis.                         |
| Input          | Execute | BOOL         | FALSE   | Starts execution at the rising edge.        |
| Output         | Done    | BOOL         | FALSE   | TRUE: Reset done                            |
|                | Busy    | BOOL         | FALSE   | TRUE: Execution of the FB is not completed. |
|                | Error   | BOOL         | FALSE   | TRUE: An error has occurred within the FB.  |
|                | ErrorID | SMC_ERROR    | 0       | An error ID is output.                      |

#### 9.2.2 SMC3\_ReinitDrive (Reinitialize Axis)

This is a function block that restarts the drive / axis. It means that the startup phase is executed again and the application cannot control the drive until bDone of the FB is set to TRUE.

##### ■ Icon



##### ■ Parameter

| Scope          | Name | Type         | Default | Description         |
|----------------|------|--------------|---------|---------------------|
| Input / output | Axis | AXIS_REF_SM3 | -       | Specifies the axis. |

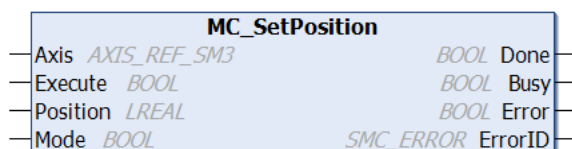
## 9.2 Motion Auxiliary Function (Change / Reset)

| Scope  | Name     | Type      | Default | Description  |
|--------|----------|-----------|---------|--|
| Input  | bExecute | BOOL      | FALSE   | Starts execution at the rising edge.                             |
|        | bVirtual | BOOL      | FALSE   | If bVirtual is set to TRUE, the axis is set to the virtual mode. |
| Output | bDone    | BOOL      | FALSE   | TRUE: Reset is completed.  |
|        | bBusy    | BOOL      | FALSE   | TRUE: Execution of the FB is not completed.                      |
|        | bError   | BOOL      | FALSE   | TRUE: An error has occurred within the FB.                       |
|        | nErrorID | SMC_ERROR | 0       | Error ID output  |

### 9.2.3 MC\_SetPosition (Change Current Position)

This is a function block (FB) that changes the current command position of the axis.

#### ■ Icon



#### ■ Parameter

| Scope          | Name     | Type         | Initial | Description  |
|----------------|----------|--------------|---------|--|
| Input / output | Axis     | AXIS_REF_SM3 | -       | Specifies the axis.  |
| Input          | Execute  | BOOL         | FALSE   | Starts execution at the rising edge.   |
|                | Position | LREAL        | 0       | Specifies the position when the mode is set to ABSOLUTE.<br>Specifies the distance when the mode is set to RELATIVE. |
|                | Mode     | BOOL         | FALSE   | TRUE: RELATIVE (Relative position)<br>FALSE: ABSOLUTE (Absolute position)  |
| Output         | Done     | BOOL         | FALSE   | TRUE: Position change is completed.  |
|                | Busy     | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.  |
|                | Error    | BOOL         | FALSE   | TRUE: An error has occurred.   |
|                | ErrorID  | SMC_ERROR    | 0       | An error ID is output.   |

## 9.3 Motion Auxiliary Function (Other Functions)

### 9.3 Motion Auxiliary Function (Other Functions)

#### 9.3.1 PMC\_ReadLatchPosition (Amplifier Latch Monitor)

This is a function block (FB) that monitor the AMP latch position. It reads the axis position when a trigger signal occurs.

##### ■ Icon



##### ■ Parameter

| Scope          | Name              | Type                     | Initial       | Description  |
|----------------|-------------------|--------------------------|---------------|--|
| Input / output | Axis              | AXIS_REF_SM3             | -             | Specifies the axis.  |
| Input          | Execute           | BOOL                     | FALSE         | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
|                | nStartCancelState | IoDRVRTEX.LATCH_STATE    | MONITOR_LATCH | Specifies the start and cancellation of the latch mode.                |
|                | nLatchTrg1        | IoDRVRTEX.LATCH_TRIGGER  | Z_PHASE       | Selects the trigger signal for latch position 1                        |
|                | nLatchTrg2        | IoDRVRTEX.LATCH_TRIGGER  | -             | Selects the trigger signal for latch position 2                        |
|                | nMonitorSel       | IoDRVRTEX.MONITOR_SELECT |               | Selects the latch position to be output as the output MonitorData.     |
| Output         | Done              | BOOL                     | FALSE         | TRUE: Output is completed.   |
|                | Busy              | BOOL                     | FALSE         | TRUE: Execution of the FB is not completed.                            |
|                | CommandAborted    | BOOL                     | FALSE         | TRUE: An interruption from other FB has occurred.                      |
|                | Error             | BOOL                     | FALSE         | TRUE: An error has occurred within the FB.                             |
|                | ErrorID           | SMC_ERROR                | 0             | An error ID is output.   |
|                | MonitorData       | LREAL                    |               | Axis position is output.   |
|                | bLatchComp1       | BOOL                     | FALSE         | TRUE : Latch completed at latch position 1 (CH1).                      |



## 9.3 Motion Auxiliary Function (Other Functions)

| Scope | Name        | Type | Initial | Description                                       |
|-------|-------------|------|---------|---|
|       | bLatchComp2 | BOOL | FALSE   | TRUE : Latch completed at latch position 2 (CH2). |

### ■ IoDRVRETEX.LATCH\_STATE (Enumeration type)

| Name               | Value | Description   |
|--------------------|-------|---|
| MONITOR_LATCH      | 80    | Monitors the position latch state.<br>Monitors the state without newly starting or canceling. |
| START_LATCH1       | 81    | Starts the position latch 1 (CH1).  |
| START_LATCH2       | 82    | Starts the position latch 2 (CH2).  |
| START_LATCH1_AND2  | 83    | Starts the position latch 1 (CH1) and position latch 2 (CH2).                                 |
| CANCEL_LATCH1      | 84    | Cancels the position latch 1 (CH1).   |
| CANCEL_LATCH2      | 88    | Cancels the position latch 2 (CH2).   |
| CANCEL_LATCH1_AND2 | 92    | Cancels the position latch 1 (CH1) and position latch 2 (CH2).                                |

### ■ IoDRVRETEX.LATCH\_TRIGGER (Enumeration type)

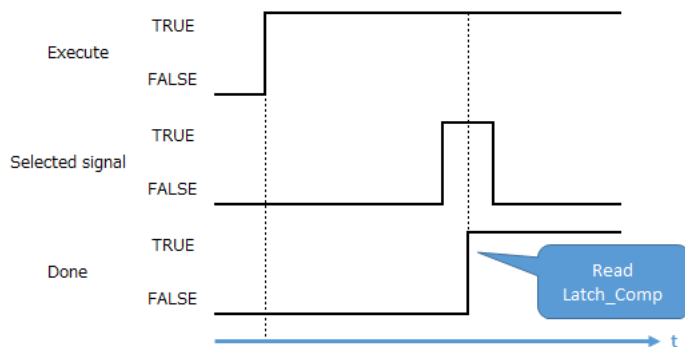
| Name                 | Value | Description           |
|----------------------|-------|-----------------------|
| Z_PHASE              | 0     | Z phase               |
| EXT1_RISING_EDGE     | 1     | Rising edge of EXT1   |
| EXT2_RISING_EDGE     | 2     | Rising edge of EXT2   |
| EXT3_RISING_EDGE     | 3     | Rising edge of EXT3   |
| PR7_111_RISING_EDGE  | 7     | Not used for this FB. |
| EXT1_FALLING_EDGE    | 9     | Falling edge of EXT1  |
| EXT2_FALLING_EDGE    | 10    | Falling edge of EXT2  |
| EXT3_FALLING_EDGE    | 11    | Falling edge of EXT3  |
| PR7_111_FALLING_EDGE | 15    | Not used for this FB. |

### ■ IoDRVRETEX.MONITOR\_SELECT (Enumeration type)

| Name  | Value | Description      |
|-------|-------|------------------|
| LPOS1 | 9     | Latch position 1 |
| LPOS2 | 10    | Latch position 2 |

## 9.3 Motion Auxiliary Function (Other Functions)

### ■ Operations when the function block is executed



The PMC\_ReadLatchPosition function block outputs the following error.

| Error                             | Description   |
|-----------------------------------|---|
| SMC_WRONG_CONTROLLER_MODE         | Executed in a mode other than the position control mode.<br>Change to SMC_position using SMC_SetControllerMode.   |
| SMC_RP_DRIVE_PARAMETER_NOT_MAPPED | Specified nLatchTrg1 and nLatchTrg2 to not use.   |
|                                   | Allocation of EXT1, EXT2, and EXT3 to the servo amplifier is faulty.<br>Change the settings for Pr4.04 to Pr4.06. |

As the PMC\_ReadLatchPosition function block uses the RTEX home return command, it cannot be executed together with PMC\_Home.

If PMC\_ReadLatchPosition is executed while PMC\_Home is being executed, the CommandAborted parameter of PMC\_ReadLatchPosition becomes TRUE.

When using EXT1, EXT2, and EXT3 for nLatchTrg1 and nLatchTrg2, set amplifier parameters as shown in the following table.

| Parameter | Parameter name      | Settings |
|-----------|---------------------|----------|
| Pr4.04    | SI5 input selection | EXT1     |
| Pr4.05    | SI6 input selection | EXT2     |
| Pr4.06    | SI7 input selection | EXT3     |

### 9.3.2 PMC\_StopLatchPosition (Stop Amplifier Latch)

This is a function block (FB) that stops the axis at the AMP latch position. Stops the axis when a trigger event occurs.

## 9.3 Motion Auxiliary Function (Other Functions)

### ■ Icon



### ■ Parameter

| Scope          | Name           | Type                    | Initial          | Description  |
|----------------|----------------|-------------------------|------------------|--|
| Input / output | Axis           | AXIS_REF_SM3            | -                | Specifies the axis.  |
| Input          | Execute        | BOOL                    | FALSE            | TRUE: Starts execution at the rising edge.<br>FALSE: Stops processing. |
|                | nLatchTrg1     | IoDRVRTEX.LATCH_TRIGGER | EXT1_RISING_EDGE | Selects the trigger signal for latch position                          |
|                | Velocity       | LREAL                   | 0                | Specifies the velocity (u/s).  |
|                | Acceleration   | LREAL                   | 0                | Specifies the acceleration (u/s <sup>2</sup> ).                        |
|                | Deceleration   | LREAL                   | 0                | Specifies the deceleration (u/s <sup>2</sup> ).                        |
|                | Jerk           | LREAL                   | 0                | Specifies the jerk (u/s <sup>3</sup> ).                                |
|                | Direction      | MC_Direction            | negative         | Specifies the traveling direction of the axis.                         |
| Output         | InVelocity     | BOOL                    | FALSE            | TRUE: The axis has reached the specified velocity for the first time.  |
|                | CommandAborted | BOOL                    | FALSE            | TRUE: An interruption from other FB has occurred.                      |
|                | Busy           | BOOL                    | FALSE            | TRUE: Execution of the FB is not completed.                            |
|                | Done           | BOOL                    | FALSE            | TRUE: Stopping is completed.   |
|                | Error          | BOOL                    | FALSE            | TRUE: An error has occurred within the FB.                             |
|                | ErrorID        | SMC_ERROR               | 0                | An error ID is output.   |
|                | MonitorData    | LREAL                   |                  | Axis position is output.   |

### ■ IoDRVRTEX.LATCH\_TRIGGER (Enumeration type)

| Name             | Value | Description           |
|------------------|-------|-----------------------|
| Z_PHASE          | 0     | Not used for this FB. |
| EXT1_RISING_EDGE | 1     | Rising edge of EXT1   |
| EXT2_RISING_EDGE | 2     | Rising edge of EXT2   |

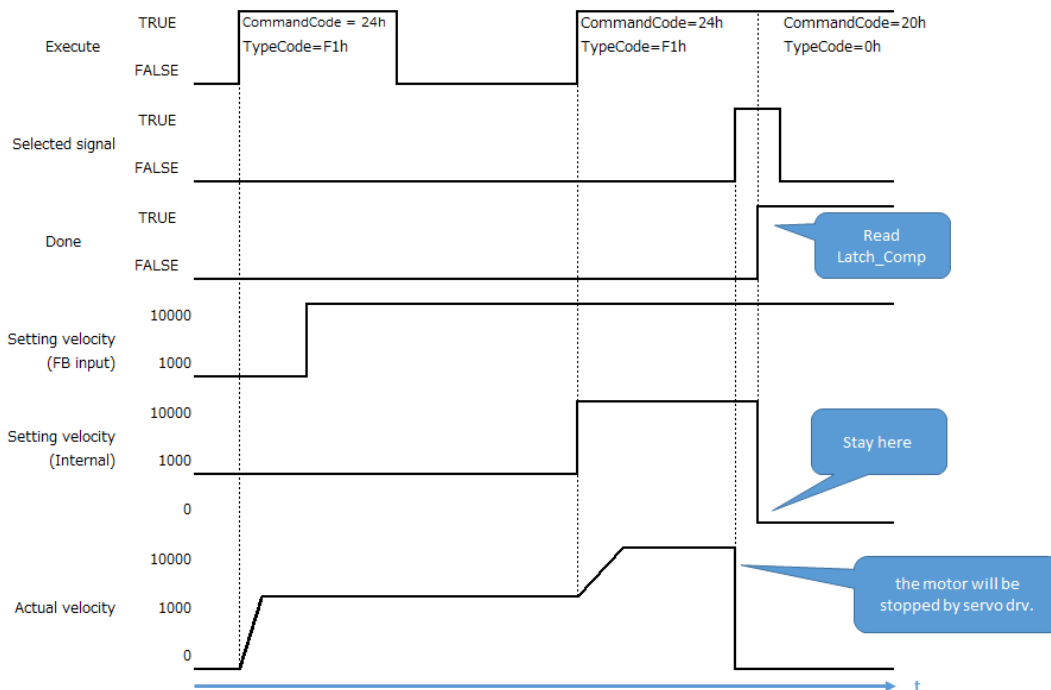
### 9.3 Motion Auxiliary Function (Other Functions)

| Name                 | Value | Description  |
|----------------------|-------|--|
| EXT3_RISING_EDGE     | 3     | Rising edge of EXT3                                |
| PR7_111_RISING_EDGE  | 7     | Condition set by MINAS amplifier parameter Pr7.111 |
| EXT1_FALLING_EDGE    | 9     | Falling edge of EXT1                               |
| EXT2_FALLING_EDGE    | 10    | Falling edge of EXT2                               |
| EXT3_FALLING_EDGE    | 11    | Falling edge of EXT3                               |
| PR7_111_FALLING_EDGE | 15    | Condition set by MINAS amplifier parameter Pr7.111 |

#### ■ MC\_Direction (Enumeration type)

| Name     | Value | Description  |
|----------|-------|--|
| positive | 1     | Travels in the positive direction.   |
| negative | -1    | Travels in the negative direction.   |
| shortest | 0     | Not available. Do not specify this.  |
| fastest  | 3     | Not available. Do not specify this.  |
| current  | 2     | Travels to the current direction.<br>Possible to use only for the modulo axis. |

#### ■ Operations when the function block is executed



- Execute = TRUE: Starts the latch mode. Execute = FALSE: Ends the latch, however, the axis operation continues as long as PMC\_StopLatchPosition is called. Stop the axis using either MC\_Stop or MC\_Halt.

## 9.3 Motion Auxiliary Function (Other Functions)

- When a trigger signal is input, the PMC\_StopLatchPosition function block ignores the command value from the GM1 and stops at the latch position.

### ■ Execution errors

The PMC\_StopLatchPosition function block outputs the following error.

| Error                                 | Description   |
|---------------------------------------|---|
| SMC_WRONG_CONTROLLER_MODE             | Executed in a mode other than the position control mode.<br>Change to SMC_position using SMC_SetControllerMode.                                       |
| SMC_RP_DRIVE_PARAMETER_NOT_MAPPED     | Specified nLatchTrg1 to not use.<br>Allocation of EXT1, EXT2, and EXT3 to the servo amplifier is faulty.<br>Change the settings for Pr4.04 to Pr4.06. |
| SMC_DI_HOMING_ERROR                   | Servo amplifier version is lower than V1.24.  |
| SMC_AXIS_NOT_READY_FOR_MOTION         | The axis is in a state (Stopping, Disabled, or Errorstop) where PMC_StopLatchPosition cannot be executed.   |
| SMC_REGULATOR_OR_START_NOT_SET        | The servo OFF or brake is applied.  |
| SMC_3SH_INVALID_VELACC_VALUES         | The input (Velocity, Acceleration, or Deceleration) is faulty.  |
| SMC_AXIS_REF_CHANGED_DURING_OPERATION | The Axis was changed during operation.  |

### ■ Execution conditions

- As the PMC\_StopLatchPosition function block uses the RTEXX home return command, it cannot be executed together with PMC\_Home.
- To use the PMC\_StopLatchPosition function block, the MINAS version must be V1.23 or higher.
- The function block supports only the control cycle of 1.0 ms and communication cycle of 0.5 ms.

### ■ Amplifier parameter conditions

When using EXT1, EXT2, and EXT3 for nLatchTrg1, set amplifier parameters as shown in the following table.

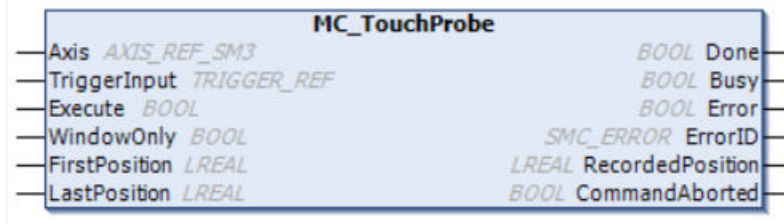
| Parameter | Parameter name      | Settings |
|-----------|---------------------|----------|
| Pr4.04    | SI5 input selection | EXT1     |
| Pr4.05    | SI6 input selection | EXT2     |
| Pr4.06    | SI7 input selection | EXT3     |

### 9.3.3 MC\_TouchProbe (Enable AMP Latch Monitoring)

This a function block (FB) that reads the axis position when a trigger signal occurs.

### 9.3 Motion Auxiliary Function (Other Functions)

#### ■ Icon



#### ■ Parameter

| Scope          | Name             | Type         | Default | Description  |
|----------------|------------------|--------------|---------|--|
| Input / output | Axis             | AXIS_REF_SM3 | -       | Specifies the axis.  |
|                | TriggerInput     | TRIGGER_REF  | 0       | Specifies the trigger signal.  |
| Input          | Execute          | BOOL         | FALSE   | TRUE: Starts execution at the rising edge.   |
|                | WindowOnly       | BOOL         | FALSE   | TRUE: The trigger event is accepted only in the specified window.                              |
|                | FirstPosition    | LREAL        | 0       | The trigger event is accepted from the start position (in the positive direction). (Unit: [u]) |
|                | LastPosition     | LREAL        | 0       | The last position up to which the trigger event is accepted. (Unit: [u])                       |
| Output         | Done             | BOOL         | FALSE   | TRUE: Halt is completed.   |
|                | Busy             | BOOL         | FALSE   | TRUE: Execution of the FB is not completed.  |
|                | Error            | BOOL         | FALSE   | TRUE: An error has occurred within the FB.   |
|                | ErrorID          | SMC_ERROR    | 0       | Error ID output  |
|                | RecordedPosition | LREAL        | 0       | The position where the trigger event has occurred. (Unit: [u])                                 |
|                | CommandAborted   | BOOL         | FALSE   | TRUE: An interruption is caused by another FB.   |

#### ■ TRIGGER\_REF (Structure)

| Member         | Type | Default | Description   |
|----------------|------|---------|---|
| iTriggerNumber | INT  | -1      | Trigger channel: Defined by the driver. (Used only when bFastLatching is TRUE.)   |
| bFastLatching  | BOOL | TRUE    | When bFastLatching is set to TRUE, latch is performed by the servo amplifier. When bFastLatching is set to FALSE, bInput is used as the trigger signal. |
| bInput         | BOOL |         | When bFastLatching is set to FALSE, the trigger signal is input.  |
| bActive        | BOOL | FALSE   | Internal variable. Do not set the value.  |

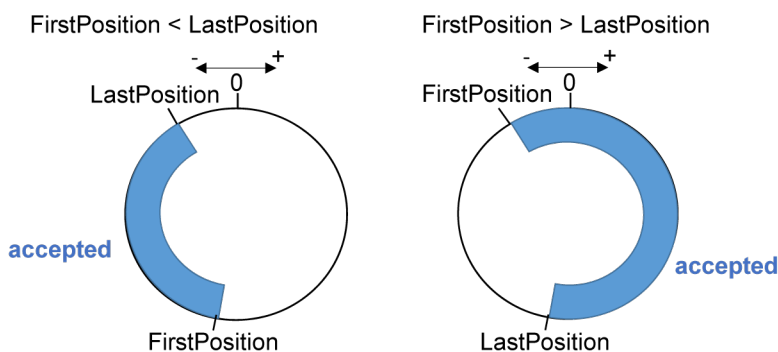
When using the MINAS, set iTriggerNumber as follows.

## 9.3 Motion Auxiliary Function (Other Functions)

| iTriggerNumber | Description          |
|----------------|----------------------|
| 0              | Rising edge of EXT1  |
| 1              | Falling edge of EXT1 |
| 2              | Rising edge of EXT2  |
| 3              | Falling edge of EXT2 |

As for pin assignment, assign EXT1 to SI5 and EXT2 to SI6.

The range where the trigger event is accepted (WindowOnly) is as follows (in case of the modulo).



Note that hardware latch (EXT1 or EXT2 trigger) is not supported. Only software latch (bInput trigger) is supported.

### 9.3.4 MC\_AbortTrigger (Disable AMP Latch Monitoring)

This is a function block (FB) that aborts the trigger event (MC\_TouchProbe).

#### ■ Icon



#### ■ Parameter

| Scope          | Name         | Type         | Default | Description                                 |
|----------------|--------------|--------------|---------|---|
| Input / output | Axis         | AXIS_REF_SM3 | -       | Specifies the axis.                         |
|                | TriggerInput | TRIGGER_REF  | 0       | Specifies the trigger signal.               |
| Input          | Execute      | BOOL         | FALSE   | TRUE: Starts execution at the rising edge.  |
| Output         | Done         | BOOL         | FALSE   | TRUE: Stopping is completed.                |
|                | Busy         | BOOL         | FALSE   | TRUE: Execution of the FB is not completed. |

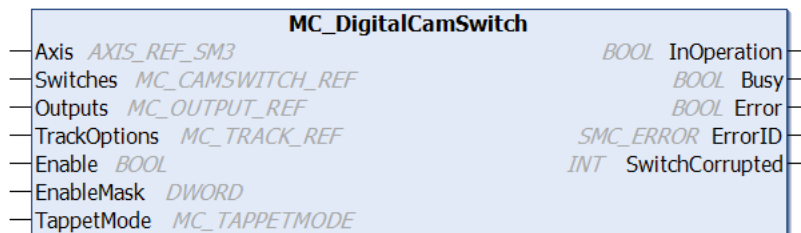
## 9.3 Motion Auxiliary Function (Other Functions)

| Scope | Name    | Type      | Default | Description                                |
|-------|---------|-----------|---------|--|
|       | Error   | BOOL      | FALSE   | TRUE: An error has occurred within the FB. |
|       | ErrorID | SMC_ERROR | 0       | Error ID output                            |

### 9.3.5 MC\_DigitalCamSwitch (Enable Digital Cam Switch)

This is a function block (FB) that performs ON / OFF control on the digital output according to the axis position. It assigns digital cam switches to tracks (maximum of 32). Switching operations can be controlled by specifying the ON / OFF position for each digital cam switch.

#### ■ Icon



#### ■ Parameter

| Scope          | Name         | Type             | Initial      | Description  |
|----------------|--------------|------------------|--------------|--|
| Input / output | Axis         | AXIS_REF_SM3     | -            | Specifies the axis where the switch is connected.  |
|                | Switches     | MC_CAMSWITCH_REF | -            | Specifies the switching operation.   |
|                | Outputs      | MC_OUTPUT_REF    | -            | ON or OFF of the switch is output. ARRAY [1..32] OF BOOL   |
|                | TrackOptions | MC_TRACK_REF     | -            | Specifies the property of the track. ARRAY [1..32] OF MC_TRACK_TR  |
| Input          | Enable       | BOOL             | FALSE        | TRUE: The FB can be executed.  |
|                | EnableMask   | DWORD            | 16#FFFFFFFF  | Specifies the track to be enabled. 1: Enabled, 0: Disabled<br>The least significant bit is the 1st track.<br>The most significant bit is the 32nd track. |
|                | TappetMode   | MC_TAPPETMODE    | tp_mode_auto | Specifies the tappet mode.   |
| Output         | InOperation  | BOOL             | FALSE        | TRUE: The track is enabled.  |
|                | Busy         | BOOL             | FALSE        | TRUE: Execution of the FB is not completed.  |
|                | Error        | BOOL             | FALSE        | TRUE: An error has occurred within the FB.   |



## 9.3 Motion Auxiliary Function (Other Functions)

| Scope | Name            | Type      | Initial | Description   |
|-------|-----------------|-----------|---------|---|
|       | ErrorID         | SMC_ERROR | 0       | An error ID is output.  |
|       | SwitchCorrupted | INT       | -1      | Index output of a faulty switch<br>-1: No problem<br>0 to 31: A problem has occurred in switches 1 to 32. |

### ■ MC\_CAMSWITCH\_REF (Structure)

| Member       | Type                       | Description   |
|--------------|----------------------------|---|
| NoOfSwitches | BYTE                       | Number of switches<br>Specifies the number of switches to be enabled when the FB is executed in the MC_CAMSWITCH_TR type array (1 to 32). |
| CamSwitchPtr | POINTER TO MC_CAMSWITCH_TR | Pointer to the first element of the MC_CAMSWITCH_TR type array  |

### ■ MC\_CAMSWITCH\_TR (Structure)

| Member          | Type  | Description  |
|-----------------|-------|--|
| TrackNumber     | INT   | Switch track number (1 to 32)  |
| FirstOnPosition | LREAL | Position where the switch turns ON when the axis is moving in the positive direction   |
| LastOnPosition  | LREAL | Position where the switch turns OFF when the axis is moving in the positive direction<br>Not used when CamSwitchMode is set to 1.  |
| AxisDirection   | INT   | Movement direction where the switch is enabled<br>0: Both positive and negative directions<br>1: Only positive direction<br>2: Only negative direction                               |
| CamSwitchMode   | INT   | Control method that performs switch ON / OFF control<br>0: ON and OFF are both controlled by the position.<br>1: ON is controlled by the position and OFF is controlled by the time. |
| Duration        | TIME  | Specifies the time during which the switch remains ON for when CamSwitchMode is set to 1.  |
| bOn             | BOOL  | Used within the FB.  |
| CounterOff      | INT   | Used within the FB.  |

### ■ MC\_TRACK\_REF (Structure)

| Member          | Type  | Description  |
|-----------------|-------|--|
| OnCompensation  | LREAL | Specifies the switch ON delay time in seconds.<br>When a positive value is specified, the switch turns ON later by the time specified.<br>When a negative value is specified, the switch turns ON earlier by the time specified. |
| OffCompensation | LREAL | Specifies the switch OFF delay time in seconds.  |

## 9.3 Motion Auxiliary Function (Other Functions)

| Member     | Type  | Description  |
|------------|-------|--|
|            |       | When a positive value is specified, the switch turns OFF after a delay of the time specified.<br>When a negative value is specified, the switch turns OFF earlier by the time specified. |
| Hysteresis | LREAL | Specifies the hysteresis value (position).   |

### ■ MC\_TAPPETMODE (Enumeration type)

| Name                   | Value | Description  |
|------------------------|-------|--|
| tp_mode_auto           | 0     | Automatically determined according to the state.<br>Servo ON state: Command position (fSetPosition) of the master axis<br>Servo OFF state: Actual position (fActPosition) of the master axis |
| tp_mode_demandposition | 1     | Command state (fSetPosition) of the master axis  |
| tp_mode_actualposition | 2     | Actual state (fActPosition) of the master axis   |

Regarding the method for entering defaults for variables of the MC\_CAMSWITCH\_TR type structure, refer to “Default Setting for Variables of the MC\_TP\_REF Type Structure”.

### ■ Operations when the function block is executed

The following sections shows switching operations (Outputs) of each track when the function block is executed after setting the parameter as follows. The axis is set to the modulo (modulo value: 1000).

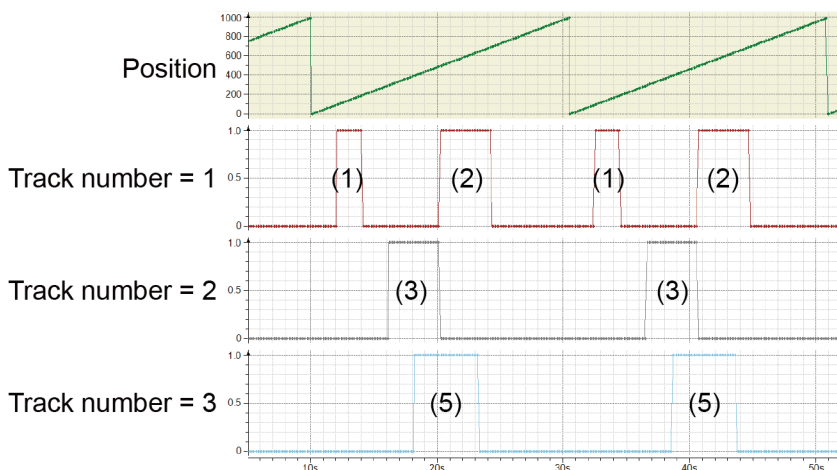
#### Function block input parameters

Five switches (CamSwitchPtr) are set.

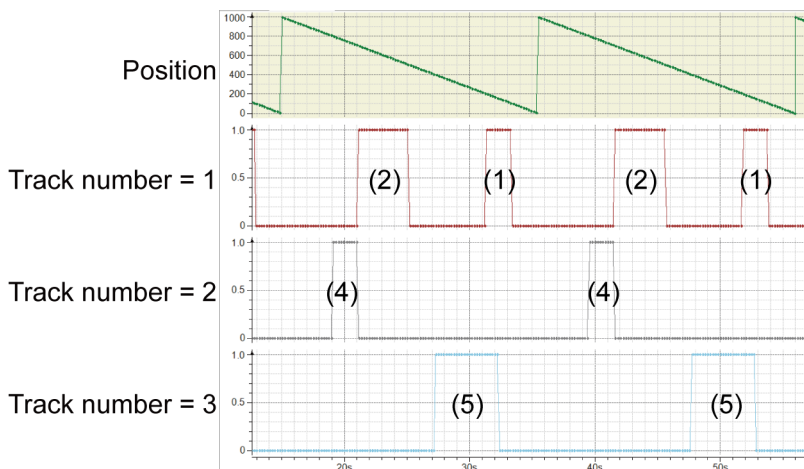
| Switch | Index | Track Number | FirstOn Position | LastOn Position | Axis Direction         | Cam SwitchMode | Duration |
|--------|-------|--------------|------------------|-----------------|------------------------|----------------|----------|
| (1)    | 1     | 1            | 100              | 200             | 0 (Both)               | 0 (Position)   | T#0ms    |
| (2)    | 2     | 1            | 500              | 700             | 0 (Both)               | 0 (Position)   | T#0ms    |
| (3)    | 3     | 2            | 300              | 500             | 1 (Positive direction) | 0 (Position)   | T#0ms    |
| (4)    | 4     | 2            | 700              | 800             | 2 (Negative direction) | 0 (Position)   | T#0ms    |
| (5)    | 5     | 3            | 400              | 0               | 0 (Both)               | 1 (Time)       | T#5s     |

#### Switching operations when the axis is moved in the positive direction

(1) to (5) are switch numbers.



### Switching operations when the axis is moved in the negative direction



#### ■ Detection of faulty switch operation (SwitchCorrupted)

SwitchCorrupted occurs when the switch does not turn ON/OFF as set.

#### — REFERENCE —

[5.6.6 Default Setting for Variables of the MC\\_TP\\_REF Type Structure](#)

### 9.3.6 SMC\_BacklashCompensation (Compensate Backlash)

This is a function block (FB) that compensates the backlash.

## 9.3 Motion Auxiliary Function (Other Functions)

### ■ Icon



### ■ Parameter

| Scope          | Name                | Type                    | Initial           | Description   |
|----------------|---------------------|-------------------------|-------------------|---|
| Input / output | Master              | AXIS_REF_SM3            | -                 | Specifies the master axis.  |
|                | Slave               | AXIS_REF_SM3            | -                 | Specifies the slave axis.   |
| Input          | bExecute            | BOOL                    | FALSE             | Starts execution at the rising edge. Remains enabled until the slave axis is interrupted by another operation or until an error occurs. |
|                | fBacklash           | LREAL                   | 0                 | Distance to compensate (backlash)   |
|                | fCompensationVel    | LREAL                   | 0                 | Additional velocity used when compensation is performed (A value to be added to the master axis velocity)                               |
|                | fCompensationAcc    | LREAL                   | 0                 | Additional acceleration used when compensation is performed (A value to be the maximum acceleration when compensation is performed)     |
|                | fCompensationDec    | LREAL                   | 0                 | Additional deceleration used when compensation is performed. (A value to be the maximum deceleration when compensation is performed)    |
|                | fCompensationJerk   | LREAL                   | 0                 | Additional jerk used when compensation is performed (Even if any value is set, the setting is disabled.)                                |
|                | eBacklashMode       | SMC_BACKLASH_MODE       | SMC_BL_AUTO       | Backlash compensation mode  |
|                | eBacklashStartState | SMC_BACKLASH_STARTSTATE | SMC_BL_START_NONE | Specifies the start conditions whether compensation is required or not when starting the backlash compensation.                         |
| Output         | bBusy               | BOOL                    | FALSE             | TRUE: The FB is in operation.   |
|                | bCommandAborted     | BOOL                    | FALSE             | TRUE: An interruption from other FB has occurred.   |

## 9.3 Motion Auxiliary Function (Other Functions)

| Scope | Name          | Type      | Initial | Description                                |
|-------|---------------|-----------|---------|--|
|       | bError        | BOOL      | FALSE   | TRUE: An error has occurred within the FB. |
|       | iErrorID      | SMC_ERROR | 0       | An error ID is output.                     |
|       | bCompensating | BOOL      | FALSE   | TRUE: Backlash compensation in operation   |

### ■ MC\_BACKLASH\_MODE (Enumeration type)

| Name            | Value | Description  |
|-----------------|-------|--|
| SMC_BL_AUTO     | 2     | Compensation in the traveling direction of the master axis |
| SMC_BL_POSITIVE | 1     | Compensation in the positive direction                     |
| SMC_BL_NEGATIVE | -1    | Compensation in the negative direction                     |
| SMC_BL_OFF      | 0     | No backlash compensation                                   |

### ■ SMC\_BACKLASH\_STARTSTATE (Enumeration type)

| Name                  | Value | Description  |
|-----------------------|-------|--|
| SMC_BL_START_NEGATIVE | -1    | If the slave axis is driven in the negative direction when compensation is started: <ul style="list-style-type: none"> <li>To make the axis travel in the positive direction, compensation is required for the backlash distance (fBacklash).</li> <li>No compensation is required for the travels in the negative direction.</li> </ul> |
| SMC_BL_START_NONE     | 0     | If the slave axis is not driven in either direction when compensation is started: To make the axis travel in the positive or negative direction, compensation is required for half the amount of the backlash distance (fBacklash).  |
| SMC_BL_START_POSITIVE | 1     | If the slave axis is driven in the positive direction when compensation is started: <ul style="list-style-type: none"> <li>No compensation is required for the travels in the positive direction.</li> <li>To make the axis travel in the negative direction, compensation is required for the backlash distance (fBacklash).</li> </ul> |



- When starting operation, make sure that both the master axis and slave axis are in the same position. If they are not set at the same position, the slave axis travels to the master axis position at the moment when SMC\_BacklashCompensation is executed.
- SMC\_BacklashCompensation functions in the same way as the phase synchronous operation (MC\_Phasing) and the phase depends on the master axis direction.

(MEMO)

# 10 Other Function Blocks

---

|        |   |       |
|--------|---|-------|
| 10.1   | COM Port (General-purpose Communication)                                | 10-4  |
| 10.1.1 | COM.Open (Open COM port)  | 10-4  |
| 10.1.2 | COM.Close (Close COM Port)  | 10-6  |
| 10.1.3 | COM.Read (Read COM Port)  | 10-7  |
| 10.1.4 | COM.Write (Write COM Port)  | 10-8  |
| 10.1.5 | COM.ERROR (Error ID)  | 10-8  |
| 10.2   | COM port (Modbus COM)   | 10-10 |
| 10.2.1 | IoDrvModbusComPort  | 10-10 |
| 10.2.2 | IoDrvModbus.ModbusChannel(Start Sending Modbus Command)                 | 10-10 |
| 10.2.3 | IoDrvModbus.ModbusRequest (Modbus Request)                              | 10-11 |
| 10.2.4 | IoDrvModbus.ModbusRequest 2 (Modbus Request 2)                          | 10-12 |
| 10.2.5 | IoDrvModbus.ModbusSlaveComPort  | 10-14 |
| 10.2.6 | IoDrvModbus.MB_ErrorCodes (Error Codes)                                 | 10-14 |
| 10.3   | LAN port (IoDrvEthernet)  | 10-16 |
| 10.3.1 | IoDrvEthernet   | 10-16 |
| 10.3.2 | IoDrvEthernet.IPARRAY_TO_INADDR (Array Type to Union Type)              | 10-16 |
| 10.3.3 | IoDrvEthernet.IPARRAY_TO_IPSTRING (Array Type to Character String Type) | 10-17 |
| 10.3.4 | IoDrvEthernet.IPARRAY_TO_UDINT (Array Type to UDINT Type)               | 10-17 |
| 10.3.5 | IoDrvEthernet.IPSTRING_TO_UDINT (Character String Type to UDINT Type)   | 10-18 |
| 10.3.6 | IoDrvEthernet.UDINT_TO_IPARRAY (UDINT Type to Array Type)               | 10-18 |
| 10.3.7 | IoDrvEthernet.UDINT_TO_IPSTRING (UDINT Type to Character String Type)   | 10-18 |
| 10.4   | LAN Port (General-purpose Communication)                                | 10-20 |
| 10.4.1 | NBS.TCP_Client (Connect to TCP Client)                                  | 10-20 |
| 10.4.2 | NBS.TCP_Connection (Connect TCP)  | 10-20 |
| 10.4.3 | NBS.TCP_Read (Receive TCP Data)   | 10-21 |
| 10.4.4 | NBS.TCP_Server (Connect TCP Server)                                     | 10-22 |
| 10.4.5 | NBS.TCP_Write (Send TCP Data)   | 10-23 |
| 10.4.6 | NBS.UDP_Peer (Open UDP Port)  | 10-23 |
| 10.4.7 | NBS.UDP_Receive (Receive UDP Data)                                      | 10-24 |
| 10.4.8 | NBS.ERROR (Error Code)  | 10-25 |
| 10.4.9 | NBS.UDP_Send (Send UDP Data)  | 10-26 |
| 10.5   | LAN Port (Modbus TCP)   | 10-28 |
| 10.5.1 | IoDrvModbusTCP  | 10-28 |
| 10.5.2 | IoDrvModbusTCP.ModbusChannel (Start Sending Modbus Command)             | 10-28 |

## 10 Other Function Blocks

---

|         |   |       |
|---------|---|-------|
| 10.5.3  | IoDrvModbusTCP.ModbusRequest (Modbus Request)                             | 10-29 |
| 10.5.4  | IoDrvModbusTCP Slave  | 10-30 |
| 10.5.5  | IoDrvModbus.MB_ErrorCodes (Error Codes)                                   | 10-31 |
| 10.6    | LAN Port (EtherNet/IP)  | 10-33 |
| 10.6.1  | IoDrvEtherNetIP (EtherNet/IP Scanner Device)                              | 10-33 |
| 10.6.2  | RemoteAdapter (Remote Adapter Device)                                     | 10-34 |
| 10.6.3  | IoDrvEtherNetIPAdapter (EtherNet/IP adapter device)                       | 10-36 |
| 10.6.4  | Module (EtherNet/IP Module Device)  | 10-37 |
| 10.6.5  | Apply_Attributes (Apply_Attributes Service)                               | 10-38 |
| 10.6.6  | Generic_Service (Generic Service Execution)                               | 10-39 |
| 10.6.7  | Get_Attribute_Single (Inquire Specific Attributes of a Specific Instance) | 10-40 |
| 10.6.8  | Get_Attributes_All (Inquire All Attributes of a Specific Instance)        | 10-41 |
| 10.6.9  | Set_Attribute_Single (Set Specific Attributes of a Specific Instance)     | 10-42 |
| 10.6.10 | Set_Attributes_All (Set All Attributes of a Specific Instance)            | 10-43 |
| 10.6.11 | NOP (NOP Service)   | 10-44 |
| 10.6.12 | Reset (Reset Service)   | 10-45 |
| 10.6.13 | Start (Start Service)   | 10-45 |
| 10.6.14 | Stop (Stop Service)   | 10-46 |
| 10.6.15 | ENIP.ERROR (Message Service Instruction Error Code)                       | 10-47 |
| 10.6.16 | ENIP.CIPClass (Service Class Code)  | 10-49 |
| 10.7    | SD Card Operation (File Operation)  | 10-52 |
| 10.7.1  | FILE.Open (Open File)   | 10-52 |
| 10.7.2  | FILE.Close (Close File)   | 10-53 |
| 10.7.3  | FILE.Read (Read File)   | 10-53 |
| 10.7.4  | FILE.Write (Write File)   | 10-54 |
| 10.7.5  | FILE.Flush (Flush File)   | 10-55 |
| 10.7.6  | FILE.Copy (Copy File)   | 10-56 |
| 10.7.7  | FILE.Rename (Rename File)   | 10-57 |
| 10.7.8  | FILE.Delete (Delete File)   | 10-58 |
| 10.7.9  | FILE.EOF (End of File)  | 10-58 |
| 10.7.10 | FILE.GetAttribute (Get File Attribute)                                    | 10-59 |
| 10.7.11 | FILE.GetPos (Get File Offset)   | 10-60 |
| 10.7.12 | FILE.GetSize (Get File Size)  | 10-60 |
| 10.7.13 | FILE.GetTime (Get File Update Time)                                       | 10-61 |
| 10.7.14 | FILE.SetPos (Set File Offset)   | 10-62 |
| 10.7.15 | FILE.ERROR (Error ID)   | 10-62 |
| 10.8    | SD Card Operation (Directory Operation)                                   | 10-64 |
| 10.8.1  | FILE.DirCreate (Create Directory)   | 10-64 |
| 10.8.2  | FILE.DirOpen (Open Directory)   | 10-64 |
| 10.8.3  | FILE.DirClose (Close Directory)   | 10-65 |
| 10.8.4  | FILE.DirCopy (Copy Directory)   | 10-66 |
| 10.8.5  | FILE.DirRename (Rename Directory)   | 10-67 |
| 10.8.6  | FILE.DirRemove (Delete Directory)   | 10-67 |
| 10.8.7  | FILE.DirList (Directory List)   | 10-68 |
| 10.9    | Clock Setting   | 10-70 |
| 10.9.1  | SYS_GetTime (Get Time)  | 10-70 |



---

|         |  |       |
|---------|--|-------|
| 10.9.2  | SYS_SetTime (Set Time)                               | 10-70 |
| 10.9.3  | SYS_GetTimezone (Get Time Zone Information)          | 10-71 |
| 10.9.4  | SYS_SetTimezone (Set Time Zone Information)          | 10-71 |
| 10.9.5  | SYS_DateConcat (Convert from UINT Type to DATE Type) | 10-72 |
| 10.9.6  | SYS_DateSplit (Convert from DATE Type to UINT Type)  | 10-73 |
| 10.9.7  | SYS_DTConcat (Convert from UINT Type to DT Type)     | 10-73 |
| 10.9.8  | SYS_DTSplit (Convert from UINT Type to DT Type)      | 10-74 |
| 10.9.9  | SYS_GetDayOfWeek (Get Day of the Week)               | 10-75 |
| 10.9.10 | SYS_TODConcat (Convert from UINT Type to TOD Type)   | 10-75 |
| 10.9.11 | SYS_TODSplit (Convert from TOD Type to UINT Type)    | 10-76 |
| 10.9.12 | ERROR (Clock Instruction Error Code)                 | 10-77 |
| 10.10   | System Data  | 10-78 |
| 10.10.1 | SYS_GetSystemError (Get System Error)                | 10-78 |
| 10.11   | PID Control  | 10-79 |
| 10.11.1 | PD (PD Control)                                      | 10-79 |
| 10.11.2 | PID (PID Control)                                    | 10-80 |
| 10.11.3 | PID_FIXCYCLE [PID Control (Any Cycle Time)]          | 10-81 |

## 10.1 COM Port (General-purpose Communication)

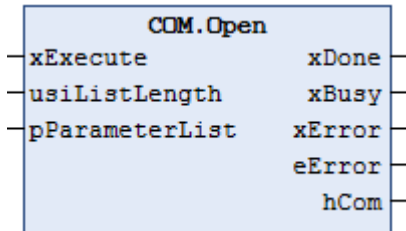
### 10.1 COM Port (General-purpose Communication)

This section describes function blocks that are used to perform general-purpose communication with the COM port.

#### 10.1.1 COM.Open (Open COM port)

This is a function block that opens a COM port. It reads from and writes to the COM port using the output handle. Close the opened COM port using the COM.Close instruction.

##### ■ Icon



##### ■ Parameter

| Scope  | Name           | Type               | Description  |
|--------|----------------|--------------------|--|
| Input  | xExecute       | BOOL               | Starts execution at the rising edge.   |
|        | usiListLength  | USINT              | Number of pParameterList entries   |
|        | pParameterList | COM.CAA.P<br>VOID  | A pointer to the communication setting parameter list for the COM port.<br>Specifies the pointer to the COM.PARAMETER structure array. |
| Output | xDone          | BOOL               | TRUE: Execution is completed.  |
|        | xBusy          | BOOL               | TRUE: Execution of the FB is not completed.  |
|        | xError         | BOOL               | TRUE: An error has occurred within the FB.   |
|        | eError         | COM.ERRO<br>R      | An error ID is output.<br>Refer to "10.1.5 COM.ERROR (Error ID)".  |
|        | hCom           | COM.CAA.<br>HANDLE | Handle of the opened COM port.   |

##### ■ COM.PARAMETER (Structure)

| Member         | Type  | Description  |
|----------------|-------|--|
| udiParameterId | UDINT | Parameter ID to be set in the COM port.<br>For a list of parameters, refer to "COM.CAA_Parameter_Constants (Constants)". |
| udiValue       | UDINT | Value to be set in the COM port  |

## 10.1 COM Port (General-purpose Communication)

### ■ COM.CAA\_Parameter\_Constants (Constants)

| Name                         | Value | Support       | Description   |
|------------------------------|-------|---------------|---|
| udiPort                      | 16#1  | Supported     | Port number (Fixed to 1.)   |
| udiStopBits                  | 16#2  | Supported     | Stop bit<br>Refer to "COM.STOPBIT (Enumeration type)".  |
| udiParity                    | 16#3  | Supported     | Parity<br>Refer to "COM.PARITY (Enumeration type)".   |
| udiBaudrate                  | 16#4  | Supported     | Baud rate (Can be selected from 9600, 19200, 38400, 57600, and 115200)                                  |
| udiTimeout                   | 16#5  | Not supported | Timeout   |
| udiBufferSize                | 16#6  | Not supported | Buffer size parameter<br>Specifies a serial buffer size.  |
| udiByteSize                  | 16#7  | Supported     | Byte size parameter<br>Sets the number of data bits to 4 to 8. (Specify 7 or 8 for the GM1 Controller.) |
| udiBinary                    | 16#8  | Not supported | Binary parameter<br>Enables the binary mode. (With the GM1 Controller, it is fixed to 0 (binary mode).) |
| udiOutxCtsFlow (Note 1)      | 16#9  | Not supported | CTS handshake for the output parameter  |
| udiOutxDsrFlow (Note 1)      | 16#A  | Not supported | DSR handshake for the output parameter  |
| udiDtrControl (Note 1)       | 16#B  | Not supported | DTR flow control parameter  |
| udiDsrSensitivity (Note 1)   | 16#C  | Not supported | DSR sensitivity parameter   |
| udiRtsControl (Note 1)       | 16#D  | Not supported | Rts flow control parameter  |
| udiTXContinueOnXoff (Note 1) | 16#E  | Not supported | XOFF continues Tx parameter.  |
| udiOutX (Note 1)             | 16#F  | Not supported | XON / XOFF output flow control parameter  |
| udiInX (Note 1)              | 16#10 | Not supported | XON / XOFF of the flow control parameter  |
| udiXonChar (Note 1)          | 16#11 | Not supported | Tx AND Rx XON character parameter   |
| udiXoffChar (Note 1)         | 16#12 | Not supported | Tx AND Rx XOFF character parameter  |
| udiXonLim (Note 1)           | 16#13 | Not supported | Sends XON threshold parameter   |
| udiXoffLim (Note 1)          | 16#14 | Not supported | Sends XOFF threshold parameter  |

(Note 1) The GM1 Controller does not support the flow control.

## 10.1 COM Port (General-purpose Communication)

### ■ COM.STOPBIT (Enumeration type)

| Name         | Value | Description                  |
|--------------|-------|------------------------------|
| ONESTOPBIT   | 0     | 1 stop bit                   |
| ONE5STOPBITS | 1     | 1.5 stop bit (Not available) |
| TWOSTOPBITS  | 2     | 2 stop bit                   |

### ■ COM.PARITY (Enumeration type)

| Name | Value | Description |
|------|-------|-------------|
| EVEN | 0     | Even        |
| ODD  | 1     | Odd         |
| NONE | 2     | None        |

### ■ Program example

#### Declaration section of ST program

```

Open : COM.Open;
OpenParam : ARRAY [1..7] OF COM.PARAMETER := [
  (udiParameterId := COM.CAA_Parameter_Constants.udiPort,      udiValue := 2),
  (udiParameterId := COM.CAA_Parameter_Constants.udiBaudrate, udiValue := 115200),
  (udiParameterId := COM.CAA_Parameter_Constants.udiParity,   udiValue := INT_TO_UDINT(COM.PARITY.ODD)),
  (udiParameterId := COM.CAA_Parameter_Constants.udiStopBits, udiValue := INT_TO_UDINT(COM.STOPBIT.ONESTOPBIT)),
  (udiParameterId := COM.CAA_Parameter_Constants.udiTimeout,  udiValue := 0),
  (udiParameterId := COM.CAA_Parameter_Constants.udiByteSize, udiValue := 8),
  (udiParameterId := COM.CAA_Parameter_Constants.udiBinary,   udiValue := 1)
];

```

#### Implementation section of ST program

```

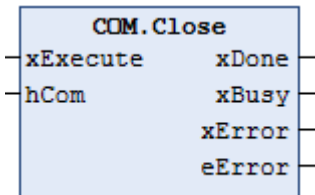
Open( xExecute:=TRUE , pParameterList:=ADR(OpenParam) , usiListLength:=SIZEOF(OpenParam)/SIZEOF(COM.PARAMETER) );

```

### 10.1.2 COM.Close (Close COM Port)

This is a function block that closes the COM port.

#### ■ Icon



#### ■ Parameter

| Scope | Name     | Type | Description                          |
|-------|----------|------|--------------------------------------|
| Input | xExecute | BOOL | Starts execution at the rising edge. |

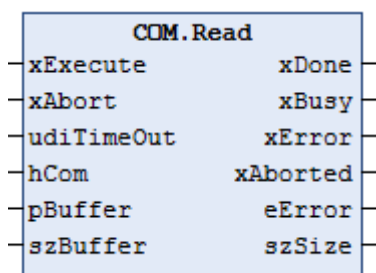
## 10.1 COM Port (General-purpose Communication)

| Scope  | Name   | Type           | Description   |
|--------|--------|----------------|---|
|        | hCom   | COM.CAA.HANDLE | Handle of the COM port to be closed                               |
| Output | xDone  | BOOL           | TRUE: Execution is completed.                                     |
|        | xBusy  | BOOL           | TRUE: Execution of the FB is not completed.                       |
|        | xError | BOOL           | TRUE: An error has occurred within the FB.                        |
|        | eError | COM.ERROR      | An error ID is output.<br>Refer to "10.1.5 COM.ERROR (Error ID)". |

### 10.1.3 COM.Read (Read COM Port)

This is a function block that reads data from the COM port.

#### ■ Icon



#### ■ Parameter

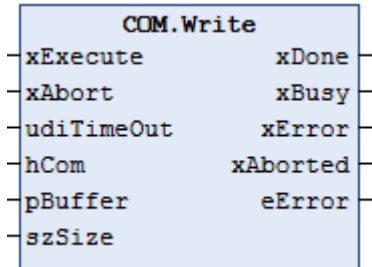
| Scope  | Name       | Type           | Description   |
|--------|------------|----------------|---|
| Input  | xExecute   | BOOL           | Starts execution at the rising edge.                              |
|        | xAbort     | BOOL           | TRUE: Stops execution and resets all outputs.                     |
|        | udiTimeOut | UDINT          | Timeout time until the execution is stopped ( $\mu$ s)            |
|        | hCom       | COM.CAA.HANDLE | Handle of the COM port  |
|        | pBuffer    | CAA.PVOID      | Pointer to the buffer that acquires data read from the COM port   |
|        | szBuffer   | CAA.SIZE       | Maximum byte of pBuffer   |
| Output | xDone      | BOOL           | TRUE: Execution is completed.                                     |
|        | xBusy      | BOOL           | TRUE: Execution of the FB is not completed.                       |
|        | xError     | BOOL           | TRUE: An error has occurred within the FB.                        |
|        | xAborted   | BOOL           | TRUE: Execution is stopped by the user.                           |
|        | eError     | COM.ERROR      | An error ID is output.<br>Refer to "10.1.5 COM.ERROR (Error ID)". |
|        | szSize     | COM.CAA.SIZE   | Data size (bytes) acquired by the pBuffer                         |

## 10.1 COM Port (General-purpose Communication)

### 10.1.4 COM.Write (Write COM Port)

This is a function block that writes data to the COM port.

#### ■ Icon



#### ■ Parameter

| Scope  | Name       | Type           | Description   |
|--------|------------|----------------|---|
| Input  | xExecute   | BOOL           | Starts execution at the rising edge.                              |
|        | xAbort     | BOOL           | TRUE: Stops execution and resets all outputs.                     |
|        | udiTimeOut | UDINT          | Timeout time until the execution is stopped (µs)                  |
|        | hCom       | COM.CAA.HANDLE | Handle of the COM port  |
|        | pBuffer    | CAA.PVOID      | Pointer to the buffer of the data written to the COM port         |
|        | szSize     | COM.CAA.SIZE   | Data size (bytes) of the pBuffer to be written to the COM port    |
| Output | xDone      | BOOL           | TRUE: Execution is completed.                                     |
|        | xBusy      | BOOL           | TRUE: Execution of the FB is not completed.                       |
|        | xError     | BOOL           | TRUE: An error has occurred within the FB.                        |
|        | xAborted   | BOOL           | TRUE: Execution is stopped by the user.                           |
|        | eError     | COM.ERROR      | An error ID is output.<br>Refer to "10.1.5 COM.ERROR (Error ID)". |

### 10.1.5 COM.ERROR (Error ID)

This is an enumeration type error ID that is output when the COM port (general-purpose communication) function block is executed.

#### ■ COM.ERROR (Enumeration type)

| Name     | Value | Description          |
|----------|-------|----------------------|
| NO_ERROR | 0     | No error             |
| TIME_OUT | 5001  | Timeout error        |
| ABORT    | 5002  | xAbort input enabled |

## 10.1 COM Port (General-purpose Communication)

---

| Name             | Value | Description      |
|------------------|-------|------------------|
| HANDLE_INVALID   | 5003  | Invalid handle   |
| ERROR_UNKNOWN    | 5004  | Unknown error    |
| WRONG_PARAMETER  | 5005  | Wrong parameter  |
| WRITE_INCOMPLETE | 5006  | Incomplete write |

## 10.2 COM port (Modbus COM)

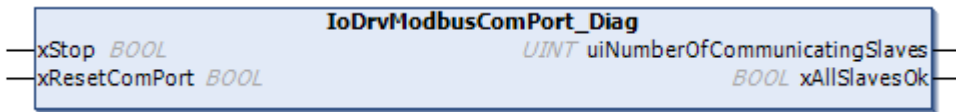
### 10.2 COM port (Modbus COM)

This section describes the instructions that are used to perform ModbusRTU communication with the COM port.

#### 10.2.1 IoDrvModbusComPort

This is a function block that controls the Modbus\_Master\_COM\_Port device.

##### ■ Icon



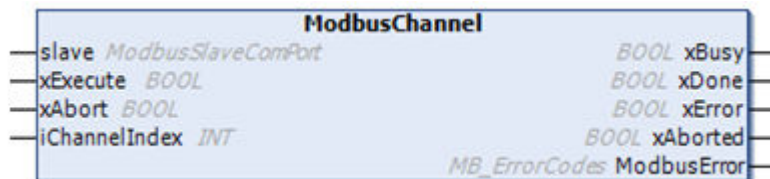
##### ■ Parameter

| Scope  | Name                          | Type | Description  |
|--------|-------------------------------|------|--|
| Input  | xStop                         | BOOL | TRUE: Stops sending a new request to the slave.<br>FALSE: Continues the current request.           |
|        | xResetComPort                 | BOOL | Closes the COM port at a rising edge.  |
| Output | uiNumberOfCommunicatingSlaves | UINT | Number of remote slaves under communication.   |
|        | xAllSlavesOk                  | BOOL | TRUE: All slaves are communicating normally.<br>FALSE: An error has occurred in one of the slaves. |

#### 10.2.2 IoDrvModbus.ModbusChannel(Start Sending Modbus Command)

This is a function block that sends the commands set in the Modbus Slave channel of the ModbusSlaveCOM\_Port device.

##### ■ Icon



##### ■ Parameter

| Scope | Name     | Type | Description                                   |
|-------|----------|------|---|
| Input | xExecute | BOOL | Starts sending commands at the rising edge.   |
|       | xAbort   | BOOL | TRUE: Stops execution and resets all outputs. |



| Scope  | Name          | Type               | Description  |
|--------|---------------|--------------------|--|
|        | iChannelIndex | INT                | Channel number where commands to be sent are set                                       |
| I/O    | slave         | ModbusSlaveComPort | Handle of the ModbusSlaveComPort device  |
| Output | xBusy         | BOOL               | TRUE: Processing of the FB is not completed.   |
|        | xDone         | BOOL               | TRUE: Processing is completed.   |
|        | xError        | BOOL               | TRUE: An error has occurred within the FB.   |
|        | xAborted      | BOOL               | TRUE: Execution is stopped by the user's xAbort input.                                 |
|        | ModbusError   | MB_ErrorCodes      | An error code is output.<br>Refer to "10.5.5 IoDrvModbus.MB_ErrorCodes (Error Codes)". |

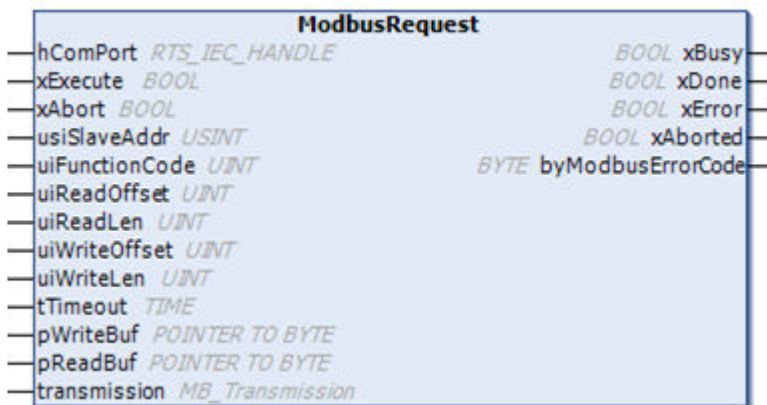
### 10.2.3 IoDrvModbus.ModbusRequest (Modbus Request)

This is a function block that processes the Modbus command specified by I/O without using the ModbusMasterComPort device.

#### ■ Supported commands

- Command 1 (Read multi-point coil state)
- Command 2 (Read multi-point input state)
- Command 3 (Read multi-point holding register)
- Command 4 (Read multi-point input register)
- Command 5 (Write single-point coil)
- Command 6 (Write single-point holding register)
- Command 15 (Write multi-point coil)
- Command 16 (Write multi-point holding register)
- Command 23 (Read / write multi-point holding register)

#### ■ Icon



## 10.2 COM port (Modbus COM)

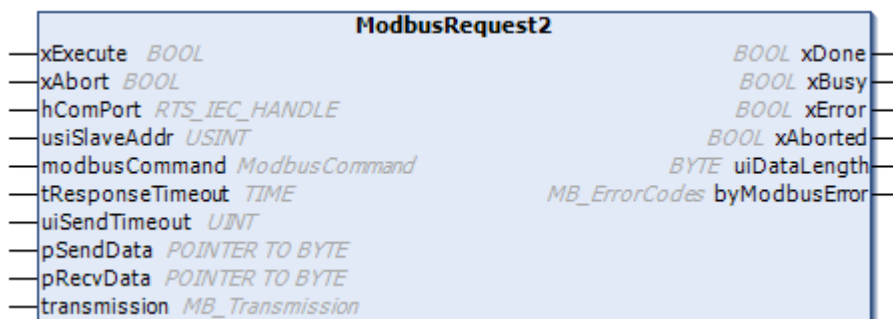
### ■ Parameter

| Scope  | Name              | Type            | Description  |
|--------|-------------------|-----------------|--|
| Input  | hComPort          | RTS_IEC_HANDLE  | COM port handle acquired by COM.Open   |
|        | xExecute          | BOOL            | Starts sending commands at the rising edge.  |
|        | xAbort            | BOOL            | TRUE: Stops execution and resets all outputs.  |
|        | usiSlaveAddr      | USINT           | Slave address 1 to 247   |
|        | uiFunctionCode    | UINT            | Modbus function code   |
|        | uiReadOffset      | UINT            | Read address offset (0 to 65535)   |
|        | uiReadLen         | UINT            | Read length (1 to 125)   |
|        | uiWriteOffset     | UINT            | Write address offset (0 to 65535)  |
|        | uiWriteLen        | UINT            | Write length (1 to 121)  |
|        | tTimeout          | UINT            | Timeout value (in ms units)  |
|        | pWriteBuf         | POINTER TO BYTE | Pointer to the send buffer.  |
|        | pReadBuf          | POINTER TO BYTE | Pointer to the receive buffer  |
|        | transmission      | MB_Transmission | Transmission type (RTU / ASCII)<br>* Supports only RTU.                                |
| Output | xBusy             | BOOL            | TRUE: Processing of the FB is not completed.   |
|        | xDone             | BOOL            | TRUE: Processing is completed.   |
|        | xError            | BOOL            | TRUE: An error has occurred within the FB.   |
|        | xAborted          | BOOL            | TRUE: Execution is stopped by the user's xAbort input.                                 |
|        | byModbusErrorCode | BYTE            | An error code is output.<br>Refer to "10.2.6 IoDrvModbus.MB_ErrorCodes (Error Codes)". |

### 10.2.4 IoDrvModbus.ModbusRequest 2 (Modbus Request 2)

This is a function block that processes, like the ModbusRequest, the Modbus command specified by I/O without using the ModbusMasterComPort device. It is different from ModbusRequest in that the structure type is used to specify the Modbus command.

## ■ Icon



## ■ Parameter

| Scope  | Name             | Type            | Description  |
|--------|------------------|-----------------|--|
| Input  | xExecute         | BOOL            | Starts sending commands at the rising edge.  |
|        | xAbsort          | BOOL            | TRUE: Stops execution and resets all outputs.  |
|        | hComPort         | RTS_IEC_HANDLE  | COM port handle acquired by COM.Open   |
|        | usiSlaveAddr     | USINT           | Slave address 1 to 247   |
|        | modbusCommand    | ModbusCommand   | Modbus command   |
|        | tResponseTimeout | TIME            | Timeout (in ms units) of the response for a request                                    |
|        | uiSendTimeout    | UINT            | Transmission timeout   |
|        | pSendData        | UINT            | Pointer to the send data   |
|        | pRecvData        | UINT            | Pointer to the receive data  |
| Output | transmission     | MB_Transmission | Transmission type (RTU / ASCII)<br>* Supports only RTU.                                |
|        | xDone            | BOOL            | TRUE: Processing is completed.   |
|        | xBusy            | BOOL            | TRUE: Processing of the FB is not completed.   |
|        | xError           | BOOL            | TRUE: An error has occurred within the FB.   |
|        | xAborted         | BOOL            | TRUE: Execution is stopped by the user's xAbort input.                                 |
|        | uiDataLength     | BYTE            | Received data length (byte)  |
|        | byModbusError    | MB_ErrorCodes   | An error code is output.<br>Refer to "10.2.6 IoDrvModbus.MB_ErrorCodes (Error Codes)". |

## ■ ModbusCommand (Structure)

| Name           | Type | Description             |
|----------------|------|-------------------------|
| uiFunctionCode | UINT | Modbus command code     |
| uiReadOffset   | UINT | Read address 0 to 65535 |

## 10.2 COM port (Modbus COM)

| Name          | Type | Description  |
|---------------|------|--|
| uiReadLen     | UINT | Range in the number of read instances varies depending on commands.  |
| uiWriteOffset | UINT | Write address 0 to 65535   |
| uiWriteLen    | UINT | Range in the number of write instances varies depending on commands. |

### 10.2.5 IoDrvModbus.ModbusSlaveComPort

This is a function block that controls the Modbus\_Slave\_COM\_Port device.

#### ■ Icon



#### ■ Parameter

| Scope  | Name          | Type          | Description   |
|--------|---------------|---------------|---|
| Input  | xTrigger      | BOOL          | Sends all the commands of the Modbus channel at the rising edge.                    |
|        | xReset        | BOOL          | Resets xError and byModbusError and resumes communication.                          |
|        | xAcknowledge  | BOOL          | Resumes communication without resetting xError and byModbusError.                   |
|        | xDoInit       | BOOL          | TRUE: Sends a slave initialization command when communication is resumed.           |
| Output | xInitDone     | BOOL          | TRUE: Modbus slave initialization command is fully completed.                       |
|        | xBusy         | BOOL          | TRUE: Processing of the FB is not completed.  |
|        | xDone         | BOOL          | TRUE: Processing is completed.  |
|        | xError        | BOOL          | TRUE: An error has occurred within the FB.  |
|        | byModbusError | MB_ErrorCodes | An error code is output. Refer to "10.2.6 IoDrvModbus.MB_ErrorCodes (Error Codes)". |
|        | iChannelIndex | INT           | Channel index   |

### 10.2.6 IoDrvModbus.MB\_ErrorCodes (Error Codes)

This is an enumeration type error code that is output when the function block for Modbus communication instruction that uses the COM port is executed.

### ■ IoDrvModbus.MB\_ErrorCodes (Enumeration type)

| Name                             | Value | Description                                |
|----------------------------------|-------|--|
| RESPONSE_SUCCESS                 | 16#0  | Succeeded                                  |
| ILLEGAL_FUNCTION                 | 16#1  | Function code not supported by the slave   |
| ILLEGAL_DATA_ADDRESS             | 16#2  | Register offset not supported by the slave |
| ILLEGAL_DATA_VALUE               | 16#3  | Illegal data writing                       |
| SLAVE_DEVICE_FAILURE             | 16#4  | Non-recoverable error                      |
| ACKNOWLEDGE                      | 16#5  | Start operation                            |
| SLAVE_DEVICE_BUSY                | 16#6  | During operation                           |
| MEMORY_PARITY_ERROR              | 16#8  | Memory parity error                        |
| GATEWAY_PATH_UNAVAILABLE         | 16#A  | Gateway path unavailable                   |
| GATEWAY_DEVICE_FAILED_TO_RESPOND | 16#B  | Gateway device failed to respond           |
| RESPONSE_TIMEOUT                 | 16#A1 | Timeout                                    |
| RESPONSE_CRC_FAIL                | 16#A2 | CRC error                                  |
| RESPONSE_WRONG_SLAVE             | 16#A3 | Wrong response                             |
| RESPONSE_WRONG_FUNCTIONCODE      | 16#A4 | Wrong function code in the response        |
| REQUEST_FAILED_TO_SEND           | 16#A5 | Request not sent                           |
| RESPONSE_INVALID_DATA            | 16#A6 | Invalid response data                      |
| RESPONSE_INVALID_PROTOCOL        | 16#A7 | Invalid response protocol                  |
| RESPONSE_INVALID_HEADER          | 16#A8 | Invalid response header                    |
| UNDEFINED                        | 16#FF | Undefined                                  |

## 10.3 LAN port (IoDrvEthernet)

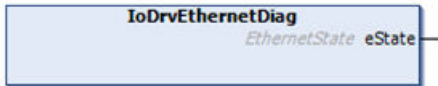
### 10.3 LAN port (IoDrvEthernet)

This section describes the library functions that are used for the network interface to perform communication with the LAN port.

#### 10.3.1 IoDrvEthernet

This is a function block that acquires the status of the LANPort device.

##### ■ Icon



##### ■ Parameter

| Scope  | Name   | Type          | Description  |
|--------|--------|---------------|--|
| Output | eState | EthernetState | Ethernet state<br>Refer to "EthernetState (Enumeration type)". |

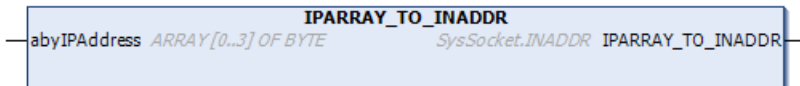
##### ■ EthernetState (Enumeration type)

| Name           | Value | Description               |
|----------------|-------|---------------------------|
| NOT_CONFIGURED | 0     | Before configuration      |
| CONFIGURED     | 1     | After configuration       |
| DISCONNECTED   | 2     | Disconnected              |
| RUNNING        | 3     | Being executed            |
| ERROR          | 4     | An error has occurred.    |
| SET_IP_ERROR   | 5     | An IP error has occurred. |

#### 10.3.2 IoDrvEthernet.IPARRAY\_TO\_INADDR (Array Type to Union Type)

This is a function that converts an array type IP address to an INADDR (union type).

##### ■ Icon



##### ■ Parameter

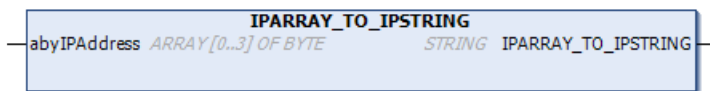
| Scope | Name           | Type                | Description      |
|-------|----------------|---------------------|------------------|
| Input | abyIPAddresses | ARRAY[0..3] OF BYTE | IP address array |

| Scope  | Name              | Type             | Description           |
|--------|-------------------|------------------|-----------------------|
| Output | IPARRAY_TO_INADDR | SysSocket.INADDR | Union type IP address |

### 10.3.3 IoDrvEthernet.IPARRAY\_TO\_IPSTRING (Array Type to Character String Type)

This is a function that converts an array type IP address to a character string type.

■ Icon



■ Parameter

| Scope  | Name              | Type                | Description                      |
|--------|-------------------|---------------------|----------------------------------|
| Input  | abyIPAddresses    | ARRAY[0..3] OF BYTE | IP address array                 |
| Output | IPARRAY_TO_STRING | STRING              | Character string type IP address |

### 10.3.4 IoDrvEthernet.IPARRAY\_TO\_UDINT (Array Type to UDINT Type)

This is a function that converts an array type IP address to a UDINT type.

■ Icon



■ Parameter

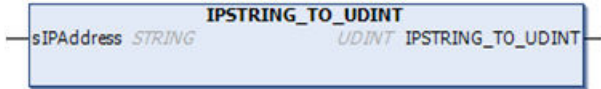
| Scope  | Name             | Type                | Description           |
|--------|------------------|---------------------|-----------------------|
| Input  | abyIPAddress     | ARRAY[0..3] OF BYTE | IP address array      |
| Output | IPARRAY_TO_UDINT | UDINT               | UDINT type IP address |

## 10.3 LAN port (IoDrvEthernet)

### 10.3.5 IoDrvEthernet.IPSTRING\_TO\_UDINT (Character String Type to UDINT Type)

This is a function that converts a character string type IP address to a UDINT type.

#### ■ Icon



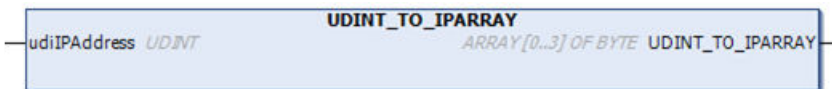
#### ■ Parameter

| Scope  | Name             | Type   | Description                      |
|--------|------------------|--------|----------------------------------|
| Input  | abyIPAddress     | STRING | Character string type IP address |
| Output | IPARRAY_TO_UDINT | UDINT  | UDINT type IP address            |

### 10.3.6 IoDrvEthernet.UDINT\_TO\_IPARRAY (UDINT Type to Array Type)

This is a function that converts a UDINT type IP address to an array type.

#### ■ Icon



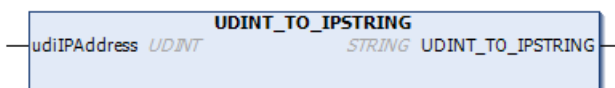
#### ■ Parameter

| Scope  | Name             | Type                | Description           |
|--------|------------------|---------------------|-----------------------|
| Input  | abyIPAddress     | UDINT               | UDINT type IP address |
| Output | UDINT_TO_IPARRAY | ARRAY[0..3] OF BYTE | IP address array      |

### 10.3.7 IoDrvEthernet.UDINT\_TO\_IPSTRING (UDINT Type to Character String Type)

This is a function that converts a UDINT type IP address to an array type.

#### ■ Icon





**■ Parameter**

| Scope  | Name            | Type   | Description                      |
|--------|-----------------|--------|----------------------------------|
| Input  | abyIPAddresses  | UDINT  | UDINT type IP address            |
| Output | UDINT_TO_STRING | STRING | Character string type IP address |

## 10.4 LAN Port (General-purpose Communication)

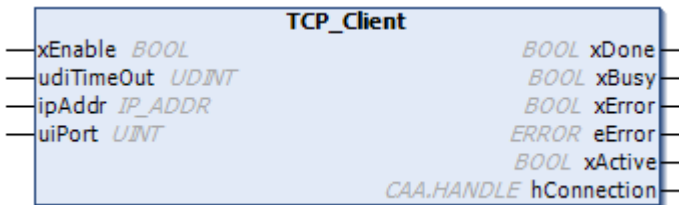
### 10.4 LAN Port (General-purpose Communication)

This section describes the library functions that are used to perform general-purpose communication with the LAN port using the TCP or UDP protocol.

#### 10.4.1 NBS.TCP\_Client (Connect to TCP Client)

This is a function block that connects to the TCP/IP client.

##### ■ Icon



##### ■ Parameter

| Scope  | Name        | Type           | Description  |
|--------|-------------|----------------|--|
| Input  | xEnable     | BOOL           | TRUE: Active   |
|        | udiTimeout  | UDINT          | Connection timeout (us)<br>No timeout when set to 0.           |
|        | ipAddr      | NBS.IP<br>ADDR | Server IP address (character string type)                      |
|        | uiPort      | UINT           | Server port No.  |
| Output | xDone       | BOOL           | TRUE: Processing is completed.                                 |
|        | xBusy       | BOOL           | TRUE: Processing of the FB is not completed.                   |
|        | xError      | BOOL           | TRUE: An error has occurred within the FB.                     |
|        | eError      | NBS.ERRO<br>R  | Connection result<br>Refer to "10.4.8 NBS.ERROR (Error Code)". |
|        | xActive     | BOOL           | TRUE: Connection is established.                               |
|        | hConnection | CAA.HAND<br>LE | Connection handle (Valid when xActive = TRUE)                  |

#### 10.4.2 NBS.TCP\_Connection (Connect TCP)

This is a function block that establishes the connection of the client connecting to the connection port opened by TCP\_Server.

### ■ Icon



### ■ Parameter

| Scope  | Name        | Type       | Description  |
|--------|-------------|------------|--|
| Input  | xEnable     | BOOL       | TRUE: Active   |
|        | hServer     | CAA.HANDLE | Connection port handle acquired by TCP_Server                  |
| Output | xDone       | BOOL       | TRUE: Processing is completed.                                 |
|        | xBusy       | BOOL       | TRUE: Processing of the FB is not completed.                   |
|        | xError      | BOOL       | TRUE: An error has occurred within the FB.                     |
|        | eError      | NBS.ERROR  | Connection result<br>Refer to "10.4.8 NBS.ERROR (Error Code)". |
|        | xActive     | BOOL       | TRUE: Connection is established. (Note 1)                      |
|        | hConnection | CAA.HANDLE | Connection handle (Valid when xActive = TRUE)                  |

(Note 1) To detect a disconnection from the client after the line is connected, it is necessary to periodically call TCP\_Read.

### **i** Info.

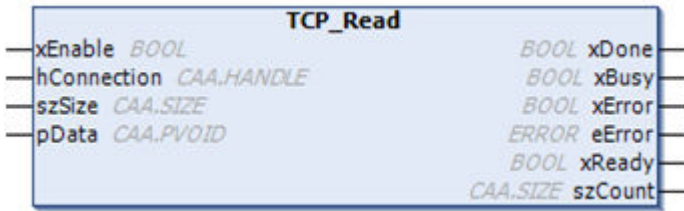
- When multiple clients are connected simultaneously to the same port, multiple TCP\_Connection instances are created.
- The hServer handle acquired by one TCP\_Server is set to the multiple TCP\_Connection instances.

### 10.4.3 NBS.TCP\_Read (Receive TCP Data)

This is a function block that acquires data received by the connection port that is established by TCP\_Connection.

## 10.4 LAN Port (General-purpose Communication)

### ■ Icon



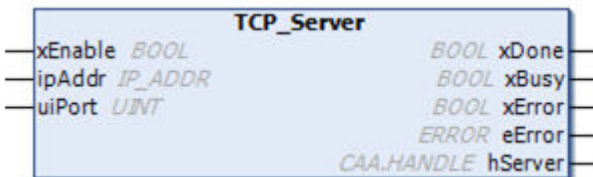
### ■ Parameter

| Scope  | Name        | Type       | Description  |
|--------|-------------|------------|--|
| Input  | xEnable     | BOOL       | TRUE: Active   |
|        | hConnection | CAA.HANDLE | Connection port handle acquired by TCP_Connection              |
|        | szSize      | CAA.SIZE   | Received buffer size (byte)                                    |
|        | pData       | CAA.PVOID  | Pointer to the receive buffer                                  |
| Output | xDone       | BOOL       | TRUE: Processing is completed.                                 |
|        | xBusy       | BOOL       | TRUE: Processing of the FB is not completed.                   |
|        | xError      | BOOL       | TRUE: An error has occurred within the FB.                     |
|        | eError      | NBS.ERROR  | Connection result<br>Refer to "10.4.8 NBS.ERROR (Error Code)". |
|        | xReady      | BOOL       | TRUE: Data is received.  |
|        | szCount     | CAA.SIZE   | Received data size (byte)                                      |

### 10.4.4 NBS.TCP\_Server (Connect TCP Server)

This is a function block that opens the specified port as a TCP/IP connection port.

### ■ Icon



### ■ Parameter

| Scope | Name    | Type        | Description   |
|-------|---------|-------------|---|
| Input | xEnable | BOOL        | TRUE: Active  |
|       | ipAddr  | NBS.IP_ADDR | Home IP address (character string), LANPort1 or LANPort2 IP address |
|       | uiPort  | UINT        | Home waiting port number  |

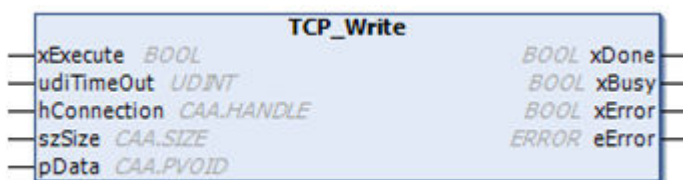
## 10.4 LAN Port (General-purpose Communication)

| Scope  | Name    | Type       | Description  |
|--------|---------|------------|--|
| Output | xDone   | BOOL       | TRUE: Processing is completed.                                 |
|        | xBusy   | BOOL       | TRUE: Processing of the FB is not completed.                   |
|        | xError  | BOOL       | TRUE: An error has occurred within the FB.                     |
|        | eError  | NBS.ERROR  | Connection result<br>Refer to "10.4.8 NBS.ERROR (Error Code)". |
|        | hServer | CAA.HANDLE | Connection handle used by TCP_Connection                       |

### 10.4.5 NBS.TCP\_Write (Send TCP Data)

This is a function block that sends data to the connection port that is established by TCP\_Connection.

#### ■ Icon



#### ■ Parameter

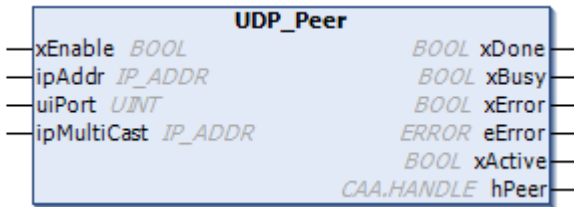
| Scope  | Name        | Type       | Description  |
|--------|-------------|------------|--|
| Input  | xExecute    | BOOL       | TRUE: Send started (edge)<br>FALSE: Processing ended (edge)    |
|        | udiTimeout  | UDINT      | Timeout (us)   |
|        | hConnection | CAA.HANDLE | Connection port handle acquired by TCP_Connection              |
|        | szSize      | CAA.SIZE   | Send data size (byte)  |
|        | pData       | CAA.PVOID  | Pointer to the send data buffer.                               |
| Output | xDone       | BOOL       | TRUE: Processing is completed.                                 |
|        | xBusy       | BOOL       | TRUE: Processing of the FB is not completed.                   |
|        | xError      | BOOL       | TRUE: An error has occurred within the FB.                     |
|        | eError      | NBS.ERROR  | Connection result<br>Refer to "10.4.8 NBS.ERROR (Error Code)". |

### 10.4.6 NBS.UDP\_Peer (Open UDP Port)

This is a function block that opens the UDP/IP port.

## 10.4 LAN Port (General-purpose Communication)

### ■ Icon



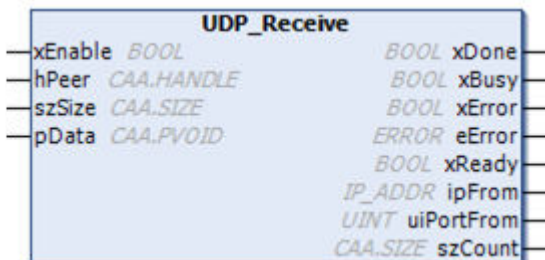
### ■ Parameter

| Scope  | Name        | Type        | Description   |
|--------|-------------|-------------|---|
| Input  | xEnable     | BOOL        | TRUE: Active<br>FALSE: Stop (xDone, xBusy, and xError are reset.)   |
|        | ipAddr      | NBS.IP_ADDR | Home IP address (character string), LANPort1 or LANPort2 IP address |
|        | uiPort      | UINT        | Home port number; Not possible to set to 0                          |
|        | ipMultiCast | NBS.IP_ADDR | Multicast address<br>("255.255.255.255"=> INADDR_NONE)              |
| Output | xDone       | BOOL        | TRUE: Processing is completed.                                      |
|        | xBusy       | BOOL        | TRUE: Processing of the FB is not completed.                        |
|        | xError      | BOOL        | TRUE: An error has occurred within the FB.                          |
|        | eError      | NBS.ERROR   | Connection result<br>Refer to "10.4.8 NBS.ERROR (Error Code)".      |
|        | xActive     | BOOL        | TRUE: Connection is established.                                    |
|        | hPeer       | CAA.HANDLE  | Connection handle (Valid when xActive = TRUE)                       |

### 10.4.7 NBS.UDP\_Receive (Receive UDP Data)

This is a function block that receives data to the connection handle acquired by `UDP_Peer`.

### ■ Icon



## 10.4 LAN Port (General-purpose Communication)

### ■ Parameter

| Scope  | Name       | Type        | Description   |
|--------|------------|-------------|---|
| Input  | xEnable    | BOOL        | TRUE: Active<br>FALSE: Stop (xDone, xBusy, and xError are reset.) |
|        | hPeer      | CAA.HANDLE  | Connection handle acquired by UDP_Peer                            |
|        | szSize     | CAA.SIZE    | Receive data buffer size (byte)                                   |
|        | pData      | CAA.PVOID   | Pointer to the receive data buffer                                |
| Output | xDone      | BOOL        | TRUE: Processing is completed.                                    |
|        | xBusy      | BOOL        | TRUE: Processing of the FB is not completed.                      |
|        | xError     | BOOL        | TRUE: An error has occurred within the FB.                        |
|        | eError     | NBS.ERROR   | Connection result<br>Refer to "10.4.8 NBS.ERROR (Error Code)".    |
|        | xReady     | BOOL        | TRUE: Data acquired, FALSE: No received data                      |
|        | ipFrom     | NBS.IP_ADDR | Data sending source IP address                                    |
|        | uiPortFrom | UINT        | Data sending source port No.                                      |
|        | szCount    | CAA.SIZE    | Received data size (byte)   |

(Note 1) If the szSize (receive data buffer size) is smaller than the received data size, only the data equivalent to the size specified by szSize is stored in pData and the data exceeding the size specified by szSize is discarded.

### 10.4.8 NBS.ERROR (Error Code)

This is an enumeration type error code that is output when the function block for communication instruction that uses the LAN port is executed.

#### ■ NBS.ERROR (Enumeration type)

| Name                  | Value | Description                      |
|-----------------------|-------|----------------------------------|
| NO_ERROR              | 0     | No error is occurring.           |
| FIRST_ERROR           | 6000  | Reserved                         |
| TIME_OUT              | 6001  | Reserved                         |
| INVALID_ADDR          | 6002  | IP address is invalid.           |
| INVALID_HANDLE        | 6003  | Handle is invalid.               |
| INVALID_DATAPOINTER   | 6004  | Data pointer is invalid.         |
| INVALID_DATASIZE      | 6005  | Data size is invalid.            |
| UDP_RECEIVE_ERROR     | 6006  | UDP datagram cannot be received. |
| UDP_SEND_ERROR        | 6007  | UDP datagram cannot be sent.     |
| UDP_SEND_NOT_COMPLETE | 6008  | Reserved                         |
| UDP_OPEN_ERROR        | 6009  | Port cannot be opened.           |

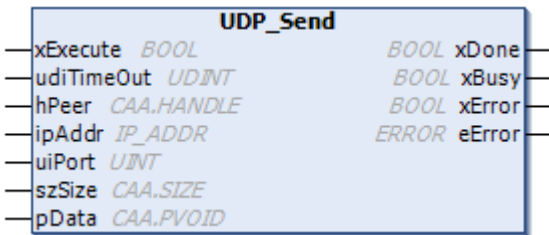
## 10.4 LAN Port (General-purpose Communication)

| Name              | Value | Description                              |
|-------------------|-------|--|
| UDP_CLOSE_ERROR   | 6010  | Port cannot be released.                 |
| TCP_SEND_ERROR    | 6011  | TCP message cannot be sent.              |
| TCP_RECEIVE_ERROR | 6012  | TCP message cannot be received.          |
| TCP_OPEN_ERROR    | 6013  | TCP port cannot be created.              |
| TCP_CONNECT_ERROR | 6014  | TCP connection cannot be established.    |
| TCP_CLOSE_ERROR   | 6015  | TCP port cannot be released.             |
| TCP_SERVER_ERROR  | 6016  | Reserved                                 |
| WRONG_PARAMETER   | 6017  | The parameter contains an invalid value. |
| ERROR_UNKNOWN     | 6018  | Reserved                                 |
| TCP_NO_CONNECTION | 6019  | There is no TCP connection.              |
| IOCTL_ERROR       | 6020  | Internal error (IOCTL is not supported.) |
| FIRST_MF          | 6050  | Reserved                                 |
| LAST_ERROR        | 6099  | Reserved                                 |

### 10.4.9 NBS.UDP\_Send (Send UDP Data)

This is a function block that sends data to the connection handle acquired by UDP\_Peer.

#### ■ Icon



#### ■ Parameter

| Scope | Name       | Type        | Description   |
|-------|------------|-------------|---|
| Input | xExecute   | BOOL        | TRUE: Send started (edge)<br>FALSE: Processing ended (edge) |
|       | udiTimeOut | UDINT       | Timeout (us)  |
|       | hPeer      | CAA.HANDLE  | Connection port handle acquired by UDP_Peer                 |
|       | ipAddr     | NBS.IP_ADDR | Destination IP address                                      |
|       | uiPort     | UINT        | Destination port No.  |
|       | szSize     | CAA.SIZE    | Send data size (byte)                                       |
|       | pData      | CAA.PVOID   | Pointer to the send data buffer.                            |



## 10.4 LAN Port (General-purpose Communication)

---

| Scope  | Name   | Type      | Description  |
|--------|--------|-----------|--|
| Output | xDone  | BOOL      | TRUE: Processing is completed.   |
|        | xBusy  | BOOL      | TRUE: Processing of the FB is not completed.                                     |
|        | xError | BOOL      | TRUE: An error has occurred within the FB.                                       |
|        | eError | NBS.ERROR | Connection result<br>Refer to " <a href="#">10.4.8 NBS.ERROR (Error Code)</a> ". |

## 10.5 LAN Port (Modbus TCP)

### 10.5 LAN Port (Modbus TCP)

This section describes the library functions that are used to perform ModbusTCP communication with the LAN port.

It is created from Modbus master TCP available in the device tree.

#### 10.5.1 IoDrvModbusTCP

This is a function block that controls the Modbus\_TCP\_Master device.

##### ■ Icon



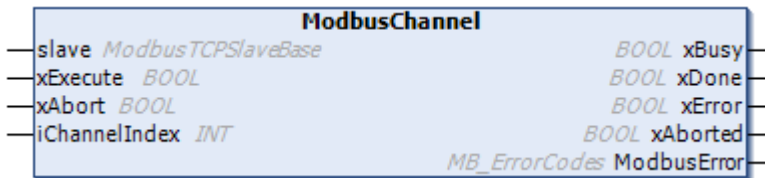
##### ■ Parameter

| Scope  | Name               | Type | Description                                |
|--------|--------------------|------|--|
| I/O    | xStop              | BOOL | TRUE: Stops sending commands to the slave. |
| Output | xSlaveError        | BOOL | There is an error in the slave function    |
|        | uiConnectes Slaves | UINT | Number of slaves connected via TCP/IP      |

#### 10.5.2 IoDrvModbusTCP.ModbusChannel (Start Sending Modbus Command)

This is a function block that sends the commands set in the Modbus Slave channel of the ModbusTCP\_Slave device.

##### ■ Icon



##### ■ Parameter

| Scope | Name           | Type                | Description                                      |
|-------|----------------|---------------------|--|
| Input | xExecute       | BOOL                | Starts sending commands at the rising edge.      |
|       | xAbort         | BOOL                | TRUE: Stops execution and resets all outputs.    |
|       | iChannellnd ex | INT                 | Channel number where commands to be sent are set |
| I/O   | slave          | ModbusTCP SlaveBase | Handle of the Modbus_TCP_Slave device<br>Output  |

| Scope  | Name        | Type          | Description  |
|--------|-------------|---------------|--|
| Output | xBusy       | BOOL          | TRUE: Processing of the FB is not completed.   |
|        | xDone       | BOOL          | TRUE: Processing is completed.   |
|        | xError      | BOOL          | TRUE: An error has occurred within the FB.   |
|        | xAborted    | BOOL          | TRUE: Execution is stopped by the user's xAbort input                                  |
|        | ModbusError | MB_ErrorCodes | An error code is output.<br>Refer to "10.5.5 IoDrvModbus.MB_ErrorCodes (Error Codes)". |

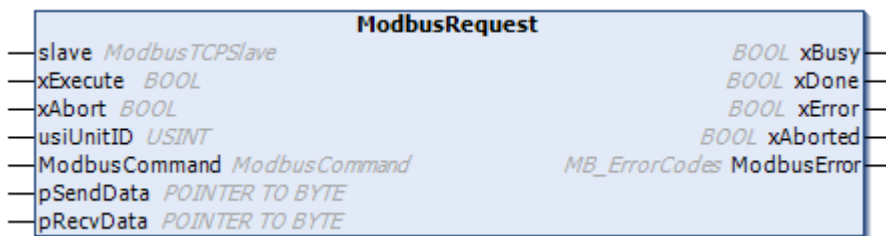
### 10.5.3 IoDrvModbusTCP.ModbusRequest (Modbus Request)

This is a function block that processes the Modbus command specified by I/O without using the Modbus\_TCP\_Slave device.

#### ■ Supported commands

- Command 1 (Read multi-point coil state)
- Command 2 (Read multi-point input state)
- Command 3 (Read multi-point holding register)
- Command 4 (Read multi-point input register)
- Command 5 (Write single-point coil)
- Command 6 (Write single-point holding register)
- Command 15 (Write multi-point coil)
- Command 16 (Write multi-point holding register)
- Command 23 (Read / write multi-point holding register)

#### ■ Icon



#### ■ Parameter

| Scope | Name          | Type            | Description  |
|-------|---------------|-----------------|--|
| Input | slave         | ModbusTCP Slave | Handle of the Modbus_TCP_Slave device                    |
|       | xExecute      | BOOL            | Starts sending commands at the rising edge.              |
|       | xAbort        | BOOL            | TRUE: Stops execution and resets all outputs.            |
|       | usiUnitID     | USINT           | Slave address 1 to 247                                   |
|       | ModbusCommand | ModbusCommand   | Structure that stores parameters of the commands issued. |

## 10.5 LAN Port (Modbus TCP)

| Scope  | Name        | Type            | Description  |
|--------|-------------|-----------------|--|
|        | pSendData   | POINTER TO BYTE | Pointer to the send data buffer.   |
|        | pRecvData   | POINTER TO BYTE | Pointer to the receive data buffer   |
| Output | xBusy       | BOOL            | TRUE: Processing of the FB is not completed.   |
|        | xDone       | BOOL            | TRUE: Processing is completed.   |
|        | xError      | BOOL            | TRUE: An error has occurred within the FB.   |
|        | xAborted    | BOOL            | TRUE: Execution is stopped by the user's xAbort input  |
|        | ModbusError | BYTE            | An error code is output.<br>Refer to "10.5.5 IoDrvModbus.MB_ErrorCodes (Error Codes)".<br>Also possible to convert the type and use as enumeration type MB_ErrorCodes. |

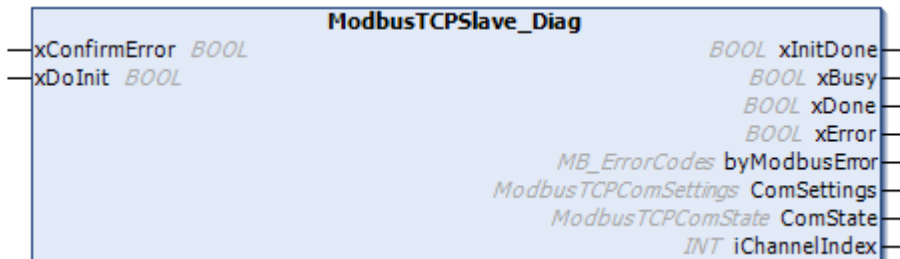
### ■ ModbusCommand (Structure)

| Name           | Type | Description  |
|----------------|------|--|
| uiFunctionCode | UINT | Modbus command code  |
| uiReadOffset   | UINT | Read address 0 to 65535  |
| uiReadLen      | UINT | Range in the number of read instances varies depending on commands.  |
| uiWriteOffset  | UINT | Write address 0 to 65535   |
| uiWriteLen     | UINT | Range in the number of write instances varies depending on commands. |

### 10.5.4 IoDrvModbusTCPSlave

This is a function block that controls the Modbus\_TCP\_Slave device.

### ■ Icon



### ■ Parameter

| Scope | Name          | Type | Description  |
|-------|---------------|------|--|
| Input | xConfirmError | BOOL | Resets xError and byModbusError and resumes communication. |

| Scope  | Name          | Type                 | Description  |
|--------|---------------|----------------------|--|
|        | xDoInit       | BOOL                 | TRUE: Sends a slave initialization command when communication is resumed.              |
| Output | xInitDone     | UINT                 | TRUE: Modbus slave initialization command is fully completed.                          |
|        | xDone         | BOOL                 | TRUE: Processing is completed.   |
|        | xBusy         | BOOL                 | TRUE: Processing of the FB is not completed.   |
|        | xError        | BOOL                 | TRUE: An error has occurred within the FB.   |
|        | byModbusError | MB_ErrorCodes        | An error code is output.<br>Refer to "10.5.5 IoDrvModbus.MB_ErrorCodes (Error Codes)". |
|        | ComSettings   | ModbusTCPComSettings | IP address and port number registered in the Modbus_TCP_Slave device.                  |
|        | ComState      | ModbusTCPComState    | Communication status   |
|        | iChannelIndex | INT                  | Channel number   |

### 10.5.5 IoDrvModbus.MB\_ErrorCodes (Error Codes)

This is an enumeration type error code that is output when the function block for Modbus communication instruction that uses the COM port is executed.

#### ■ IoDrvModbus.MB\_ErrorCodes (Enumeration type)

| Name                             | Value | Description                                |
|----------------------------------|-------|--|
| RESPONSE_SUCCESS                 | 16#0  | Succeeded                                  |
| ILLEGAL_FUNCTION                 | 16#1  | Function code not supported by the slave   |
| ILLEGAL_DATA_ADDRESS             | 16#2  | Register offset not supported by the slave |
| ILLEGAL_DATA_VALUE               | 16#3  | Illegal data writing                       |
| SLAVE_DEVICE_FAILURE             | 16#4  | Non-recoverable error                      |
| ACKNOWLEDGE                      | 16#5  | Start operation                            |
| SLAVE_DEVICE_BUSY                | 16#6  | During operation                           |
| MEMORY_PARITY_ERROR              | 16#8  | Memory parity error                        |
| GATEWAY_PATH_UNAVAILABLE         | 16#A  | Gateway path unavailable                   |
| GATEWAY_DEVICE_FAILED_TO_RESPOND | 16#B  | Gateway device failed to respond           |
| RESPONSE_TIMEOUT                 | 16#A1 | Timeout                                    |
| RESPONSE_CRC_FAIL                | 16#A2 | CRC error                                  |
| RESPONSE_WRONG_SLAVE             | 16#A3 | Wrong response                             |
| RESPONSE_WRONG_FUNCTIONCODE      | 16#A4 | Wrong function code in the response        |
| REQUEST_FAILED_TO_SEND           | 16#A5 | Request not sent                           |

## 10.5 LAN Port (Modbus TCP)

---

| Name                      | Value | Description               |
|---------------------------|-------|---------------------------|
| RESPONSE_INVALID_DATA     | 16#A6 | Invalid response data     |
| RESPONSE_INVALID_PROTOCOL | 16#A7 | Invalid response protocol |
| RESPONSE_INVALID_HEADER   | 16#A8 | Invalid response header   |
| UNDEFINED                 | 16#FF | Undefined                 |

## 10.6 LAN Port (EtherNet/IP)

This section describes the instructions that are used to control EtherNet/IP scanner and adapter functions using the SMC.

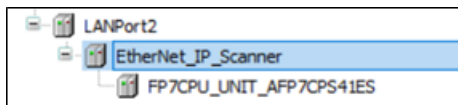
### 10.6.1 IoDrvEtherNetIP (EtherNet/IP Scanner Device)

This is a function block (FB) that controls the EtherNet/IP scanner device.

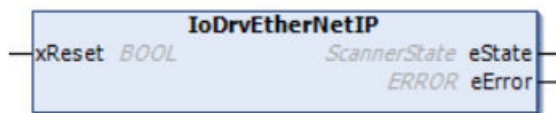
This function block is automatically generated by adding an EtherNet/IP scanner device and the name of the device that is added is used as the instance name.

#### Example

Adding an EtherNet/IP scanner device named "EtherNet\_IP\_Scanner" to LANPort2



#### ■ Icon



#### ■ Parameter

| Scope  | Name   | Type         | Description                                    |
|--------|--------|--------------|--|
| Input  | xReset | BOOL         | Resets the scanner function at the rising edge |
| Output | eState | ScannerState | EtherNet/IP scanner device state               |
|        | eError | ERROR        | Error state code of EtherNet/IP scanner        |

#### ■ ScannerState (EtherNet/IP scanner device state)

| Name                 | Description  |
|----------------------|--|
| INITIALIZING         | The device is setting up a CIP object. It is continuing IP_CONFIG.                                       |
| IP_CONFIG            | The device creates an IP configuration for Ethernet interface and waits until it enters a RUNNING state. |
| UDP_CONFIG           | The device opens the socket for UDP default port 2222.   |
| ENCAPSULATION_CONFIG | The encapsulation server for the scanner is started via the default TCP port (44818).                    |
| ADAPTER_CONFIG       | The device is in an empty state. It is continuing OPEN_CONNECTIONS.                                      |
| OPEN_CONNECTIONS     | The CIP ID status is set to "configured" and the RUNNING state continues.                                |
| RUNNING              | The device opens a connection to the adapter and processes explicit messages with I/O communication.     |

## 10.6 LAN Port (EtherNet/IP)

---

| Name                 | Description  |
|----------------------|--|
| DIAGNOSTIC_AVAILABLE | There are diagnostic messages from the configurator or editor.                                 |
| BUS_ERROR            | The UDP or TCP port failed to open.  |
| RESET                | xReset for the CIP ID object was received.   |
| ERROR                | When the network interface is in a continued state, the scanner enters the INITIALIZING state. |

### ■ ERROR (Error state code of EtherNet/IP scanner)

| Name                         | Description                          |
|------------------------------|--------------------------------------|
| NO_ERROR                     | No error is occurring.               |
| INVALID_COMMAND              | The command is invalid.              |
| OUT_OF_MEMORY                | A memory shortage occurred.          |
| INVALID_DATA                 | The data is invalid.                 |
| INVALID_SESSION_HANDLE       | The session handle is invalid.       |
| INVALID_LENGTH               | The data length is invalid.          |
| UNSUPPORTED_PROTOCOL_VERSION | The protocol version is unsupported. |
| NBS_ERROR                    | An NBS error occurred.               |
| NBS_RCV_ERROR                | Data cannot be received via NBS.     |
| NBS_SND_ERROR                | Data cannot be sent via NBS.         |
| ENCAPSULATION_ERROR          | An encapsulation error occurred.     |
| TCPIP_CONFIG_ERROR           | TCP IP settings are incorrect.       |
| UDP_CONFIG_ERROR             | UDP settings are incorrect.          |
| UDP_RECV_ERROR               | UDP datagrams cannot be received.    |
| UDP_SEND_ERROR               | UDP datagrams cannot be sent.        |
| UDP_CLOSE_ERROR              | UDP ports cannot be released.        |
| NULL_POINTER                 | This is a null pointer.              |
| DEVICE_STATE_ERROR           | An error is occurring on the device. |
| RECONFIGURATION_FAILED       | Reconfiguration failed.              |
| PERFORMANCE_MONITOR_DISABLED | The performance monitor is disabled. |
| INVALID_MEASURING_POINT      | Measuring points are invalid.        |
| IP_CONFIG_ERROR              | IP settings are faulty.              |

### 10.6.2 RemoteAdapter (Remote Adapter Device)

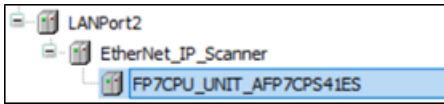
This is a function block (FB) for the remote adapter device linked to the EtherNet/IP scanner device.

This function block is automatically generated by adding an EtherNet/IP remote adapter device and the name of the device that is added is used as the instance name.

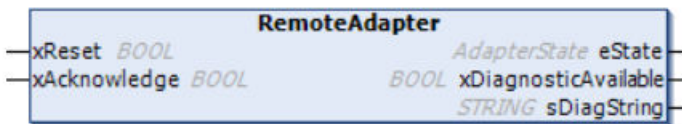


**Example**

Adding a remote adapter device named "FP7CPU\_UNIT\_AFP7CPS41ES" to EtherNet\_IP\_Scanner



■ **Icon**



■ **Parameter**

| Scope  | Name                  | Type          | Description  |
|--------|-----------------------|---------------|--|
| Input  | xReset                | BOOL          | Resets the remote adapter function at the rising edge        |
|        | xAcknowledge          | BOOL          | Acknowledges the diagnostic information at the rising edge   |
| Output | eState                | Adapter State | Remote adapter state   |
|        | xDiagnostic Available | BOOL          | The output remains TRUE when there is diagnostic information |
|        | sDiagString           | STRING        | Diagnosis string   |

■ **AdapterState (Adapter device state)**

| Name                 | Description  |
|----------------------|--|
| DISABLED             | The device is disabled in device tree                                      |
| NOT_CONFIGURED       | Parameters are being loaded  |
| IP_CONFIG            | The device has configured a TCP object and is waiting for an Ethernet node |
| ENCAPSULATION_CONFIG | Encapsulation is being configured  |
| LIST_SERVICES        | List services are being executed   |
| REGISTER_SESSION     | Register session is in progress  |
| PARAMETER_CONFIG     | Parameters are being configured  |
| CONFIGURED           | The device is in configuration completion state                            |
| RUNNING              | The device is in running state   |
| IDLE                 | The device is in idle state  |
| RESET                | UDP and TCP connection is closing  |
| RESET_SERVICE        | Reset service is being executed  |
| CONNECTIVITY_CHECK   | Connectivity check is in progress  |
| BUS_ERROR            | Bus error is occurring   |

## 10.6 LAN Port (EtherNet/IP)

| Name  | Description        |
|-------|--------------------|
| ERROR | Error is occurring |

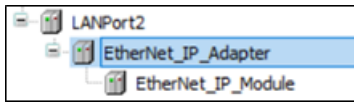
### 10.6.3 IoDrvEtherNetIPAdapter (EtherNet/IP adapter device)

This is a function block (FB) that controls the EtherNet/IP adapter device.

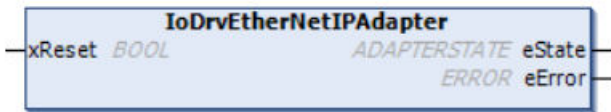
This function block is automatically generated by adding an EtherNet/IP adapter device and the name of the device that is added is used as the instance name.

#### Example

Adding an EtherNet/IP adapter device named "EtherNet\_IP\_Adapter" to LANPort2



#### ■ Icon



#### ■ Parameter

| Scope  | Name   | Type         | Description                                    |
|--------|--------|--------------|--|
| Input  | xReset | BOOL         | Resets the adapter function at the rising edge |
| Output | eState | ADAPTERSTATE | EtherNet/IP adapter device state               |
|        | eError | ERROR        | Error state code of EtherNet/IP adapter        |

#### ■ ADAPTERSTATE (EtherNet/IP adapter device state)

| Name                     | Description  |
|--------------------------|--|
| UPDATE_CONFIGURATION     | Startup phase  |
| NOT_CONFIGURED           | Parameters are being loaded  |
| DISABLED                 | The device is disabled in device tree                                      |
| CONFIGURED               | A CIP object has been created  |
| IP_CONFIG                | The device has configured a TCP object and is waiting for an Ethernet node |
| IMPLICITMESSAGING_CONFIG | UDP port has been opened   |
| EXPLICITMESSAGING_CONFIG | TCP port has been opened   |
| NO_CONNECTION            | The protocol stack has been started, but the scanner is unconnected.       |
| RUNNING                  | The protocol stack is running, and the scanner is connected.               |

| Name              | Description  |
|-------------------|--|
| STOPPED           | The Ethernet node is inactive, and the device is waiting for the Ethernet node to return.        |
| RESET             | UDP and TCP connection is closing.   |
| SCANNER_EXTENSION | If the scanner registered this adapter as an I/O extension, the adapter is active in this state. |
| ERROR             | Critical error   |
| BUS_ERROR         | Ethernet is not ready yet or is unavailable.   |

### ■ ERROR (EtherNet/IP adapter error state)

| Name                            | Description   |
|---------------------------------|---|
| NO_ERROR                        | No error  |
| TIME_OUT                        | Timeout   |
| CONFIGURATION_FAILED            | Failed to initialize resources, load connector parameters, or communicate with sub-connectors (modules) |
| IP_CONFIG_FAILED                | The Ethernet node issued an error   |
| IMPLICITMESSAGING_CONFIG_FAILED | Failed to create UDP port<br>"CIP_ENC.ParameterList.gc_uiUDPPort" (default: 2222)                       |
| EXPLICITMESSAGING_CONFIG_FAILED | Failed to create TCP / UDP port<br>"IP_ENC.ParameterList.gc_uiTCPPort" (default: 44818)                 |
| EXPLICITMESSAGE_RECEIVE_FAILED  | Problem related to TCP or UDP port socket  <br>CIP_ENC.ParameterList.gc_uiTCPPort   (default: 44818)    |
| EXPLICITMESSAGE_SEND_FAILED     | Problem related to TCP or UDP port socket  <br>CIP_ENC.ParameterList.gc_uiTCPPort   (default: 44818)    |
| LICENSE_MISSING                 | No license  |

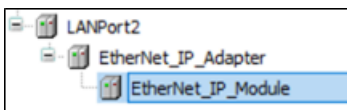
### 10.6.4 Module (EtherNet/IP Module Device)

This is a function block (FB) that controls the EtherNet/IP module device.

This function block is automatically generated by adding an EtherNet/IP module device and the name of the device that is added is used as the instance name.

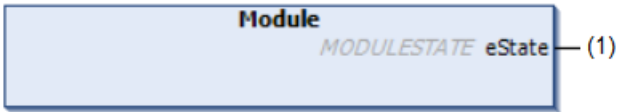
#### Example

Adding an EtherNet/IP module device named "EtherNet\_IP\_Module" to EtherNet/IP adapter device



## 10.6 LAN Port (EtherNet/IP)

### ■ Icon



### ■ Parameter

| Scope  | Name   | Type        | Description         |
|--------|--------|-------------|---------------------|
| Output | eState | MODULESTATE | Module device state |

### ■ MODULESTATE (EtherNet/IP module device state)

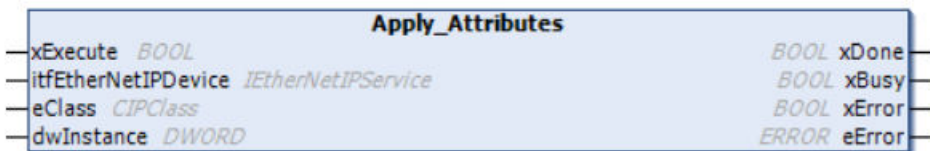
| Name           | Description   |
|----------------|---|
| NOT_CONFIGURED | Parameters are being loaded.  |
| CONFIGURED     | A CIP object has been created.  |
| NO_CONNECTION  | The protocol stack has been started, but the scanner is unconnected.                      |
| RUNNING        | The protocol stack is running, and the scanner is connected.                              |
| STOPPED        | The Ethernet node is inactive, and the device is waiting for the Ethernet node to return. |
| DISABLED       | The device is disabled in device tree.  |
| ERROR          | Critical error  |

## 10.6.5 Apply\_Attributes (Apply\_Attributes Service)

This is a function block (FB) that calls the "Apply\_Attributes" service of the CIP object instance.

The attribute set in "Get\_Attribut\_Single" or "Get\_Attribut\_All" is adopted and saved in the adapter.

### ■ Icon



### ■ Parameter

| Scope | Name                | Type               | Description  |
|-------|---------------------|--------------------|--|
| Input | xExecute            | BOOL               | Execution flag   |
|       | itfEtherNetIPDevice | IEtherNetIPService | EtherNet/IP device that implements the EtherNet/IP service |

| Scope  | Name       | Type          | Description   |
|--------|------------|---------------|---|
|        | eClass     | ENIP.CIPClass | Class that executes the service   |
|        | dwInstance | DWORD         | Instance that executes the service<br>(0: Class level, 1-x: Instance level) |
| Output | xDone      | BOOL          | Completion flag   |
|        | xBusy      | BOOL          | Busy flag   |
|        | xError     | BOOL          | Error flag  |
|        | eError     | ENIP.ERROR    | Error<br>(0-255: CIP error, 256-x: Library error)                           |

### 10.6.6 Generic\_Service (Generic Service Execution)

This is a function block (FB) that executes generic services with the EtherNet/IP adapter. Messages are sent as unconnected explicit message requests.

#### Note

- The endianness of data to be sent or received must be exchanged by devices.

#### Icon



#### Parameter

| Scope | Name                | Type               | Description   |
|-------|---------------------|--------------------|---|
| Input | xExecute            | BOOL               | Execution flag  |
|       | itfEtherNetIPDevice | IEtherNetIPService | EtherNet/IP device that implements the EtherNet/IP service                  |
|       | eClass              | ENIP.CIPClass      | Class that executes the service   |
|       | dwInstance          | DWORD              | Instance that executes the service<br>(0: Class level, 1-x: Instance level) |
|       | wAttribute          | WORD               | Attribute corresponding to the service                                      |

## 10.6 LAN Port (EtherNet/IP)

| Scope  | Name                | Type                  | Description  |
|--------|---------------------|-----------------------|--|
|        | eService            | ENIP.CIPCommonService | CIPCommonService member service code or vendor-specific service code   |
|        | pWriteData          | POINTER TO BYTE       | Pointer to data to be written to the EtherNet/IP adapter. The parameter is set to 0 when no data is sent.                  |
|        | udiWriteDataSize    | UDINT                 | Size of data to be written to the EtherNet/IP adapter. The parameter is set to 0 when no data is sent.                     |
|        | pReadData           | POINTER TO BYTE       | Storage pointer to data received from the EtherNet/IP adapter. The parameter is set to 0 when no data is received.         |
|        | udiReadDataSize     | UDINT                 | Size of storage buffer for data received from the EtherNet/IP adapter. The parameter is set to 0 when no data is received. |
| Output | xDone               | BOOL                  | Completion flag  |
|        | xBusy               | BOOL                  | Busy flag  |
|        | xError              | BOOL                  | Error flag   |
|        | eError              | ENIP.ERROR            | Error<br>(0-255: CIP error, 256-x: Library error)  |
|        | udiReceivedDataSize | UDINT                 | Size of received data  |

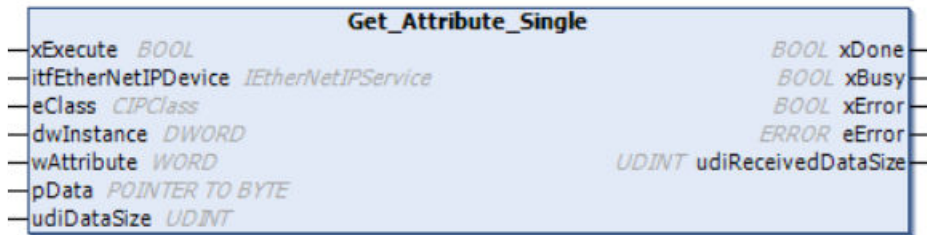
### ■ ENIP.CIPCommonService (CIPCommonService member service code)

| Name                 | Value |
|----------------------|-------|
| None                 | 16#0  |
| GET_ATTRIBUTES_ALL   | 16#1  |
| SET_ATTRIBUTES_ALL   | 16#2  |
| RESET                | 16#5  |
| START                | 16#6  |
| STOP                 | 16#7  |
| APPLY_ATTRIBUTES     | 16#D  |
| GET_ATTRIBUTE_SINGLE | 16#E  |
| SET_ATTRIBUTE_SINGLE | 16#10 |
| NO_OPERATION         | 16#17 |

### 10.6.7 Get\_Attribute\_Single (Inquire Specific Attributes of a Specific Instance)

This is a function block (FB) that inquires specific attributes of a specific instance of the CIP object.

■ Icon



■ Parameter

| Scope  | Name                | Type               | Description  |
|--------|---------------------|--------------------|--|
| Input  | xExecute            | BOOL               | Execution flag   |
|        | itfEtherNetIPDevice | IEtherNetIPService | EtherNet/IP device that implements the EtherNet/IP service               |
|        | eClass              | ENIP.CIPClass      | Class that executes the service  |
|        | dwInstance          | DWORD              | Instance that executes the service (0: Class level, 1-x: Instance level) |
|        | wAttribute          | WORD               | Attribute corresponding to the service                                   |
|        | pData               | POINTER TO BYTE    | Storage pointer to data received from the EtherNet/IP adapter            |
|        | udiDataSize         | UDINT              | Size of storage buffer for data received from the EtherNet/IP adapter    |
| Output | xDone               | BOOL               | Completion flag  |
|        | xBusy               | BOOL               | Busy flag  |
|        | xError              | BOOL               | Error flag   |
|        | eError              | ENIP.ERROR         | Error (0-255: CIP error, 256-x: Library error)                           |
|        | udiReceivedDataSize | UDINT              | Size of received data  |

**10.6.8 Get\_Attributes\_All (Inquire All Attributes of a Specific Instance)**

This is a function block (FB) that inquires all attributes of a specific instance of the CIP object.

## 10.6 LAN Port (EtherNet/IP)

### ■ Icon



### ■ Parameter

| Scope  | Name                | Type               | Description   |
|--------|---------------------|--------------------|---|
| Input  | xExecute            | BOOL               | Execution flag  |
|        | itfEtherNetIPDevice | IEtherNetIPService | EtherNet/IP device that implements the EtherNet/IP service                  |
|        | eClass              | ENIP.CIPClass      | Class that executes the service   |
|        | dwInstance          | DWORD              | Instance that executes the service<br>(0: Class level, 1-x: Instance level) |
|        | pData               | POINTER TO BYTE    | Storage pointer to data received from the EtherNet/IP adapter               |
|        | udiDataSize         | UDINT              | Size of storage buffer for data received from the EtherNet/IP adapter       |
| Output | xDone               | BOOL               | Completion flag   |
|        | xBusy               | BOOL               | Busy flag   |
|        | xError              | BOOL               | Error flag  |
|        | eError              | ENIP.ERROR         | Error<br>(0-255: CIP error, 256-x: Library error)                           |
|        | udiReceivedDataSize | UDINT              | Size of received data   |

### 10.6.9 Set\_Attribute\_Single (Set Specific Attributes of a Specific Instance)

This is a function block (FB) that sets specific attributes of a specific instance of the CIP object

### ■ Icon





■ Parameter

| Scope  | Name                 | Type                 | Description  |
|--------|----------------------|----------------------|--|
| Input  | xExecute             | BOOL                 | Execution flag   |
|        | itfEtherNet/IPDevice | IEtherNet/IP Service | EtherNet/IP device that implements the EtherNet/IP service               |
|        | eClass               | ENIP.CIPClasses      | Class that executes the service  |
|        | dwInstance           | DWORD                | Instance that executes the service (0: Class level, 1-x: Instance level) |
|        | wAttribute           | WORD                 | Attribute corresponding to the service                                   |
|        | pData                | POINTER TO BYTE      | Pointer to data to be written  |
|        | udiDataSize          | UDINT                | Size of data to be written   |
| Output | xDone                | BOOL                 | Completion flag  |
|        | xBusy                | BOOL                 | Busy flag  |
|        | xError               | BOOL                 | Error flag   |
|        | eError               | ENIP.ERROR           | Error (0-255: CIP error, 256-x: Library error)                           |

**10.6.10 Set\_Attributes\_All (Set All Attributes of a Specific Instance)**

This is a function block (FB) that sets all attributes of a specific instance of the CIP object.

■ Icon



■ Parameter

| Scope | Name                 | Type                 | Description  |
|-------|----------------------|----------------------|--|
| Input | xExecute             | BOOL                 | Execution flag   |
|       | itfEtherNet/IPDevice | IEtherNet/IP Service | EtherNet/IP device that implements the EtherNet/IP service               |
|       | eClass               | ENIP.CIPClasses      | Class that executes the service  |
|       | dwInstance           | DWORD                | Instance that executes the service (0: Class level, 1-x: Instance level) |

## 10.6 LAN Port (EtherNet/IP)

| Scope  | Name        | Type            | Description                                       |
|--------|-------------|-----------------|---|
|        | pData       | POINTER TO BYTE | Pointer to data to be written                     |
|        | udiDataSize | UDINT           | Size of data to be written                        |
| Output | xDone       | BOOL            | Completion flag                                   |
|        | xBusy       | BOOL            | Busy flag   |
|        | xError      | BOOL            | Error flag  |
|        | eError      | ENIP.ERROR      | Error<br>(0-255: CIP error, 256-x: Library error) |

### 10.6.11 NOP (NOP Service)

This is a function block (FB) that executes the NOP service of a specific instance of the CIP object.

Normally, this service is used to check whether the adapter can still be used in the network.

#### ■ Icon



#### ■ Parameter

| Scope  | Name                | Type               | Description   |
|--------|---------------------|--------------------|---|
| Input  | xExecute            | BOOL               | Execution flag  |
|        | itfEtherNetIPDevice | IETHERNETIPService | EtherNet/IP device that implements the EtherNet/IP service                  |
|        | eClass              | ENIP.CIPClass      | Class that executes the service   |
|        | dwInstance          | DWORD              | Instance that executes the service<br>(0: Class level, 1-x: Instance level) |
| Output | xDone               | BOOL               | Completion flag   |
|        | xBusy               | BOOL               | Busy flag   |
|        | xError              | BOOL               | Error flag  |
|        | eError              | ENIP.ERROR         | Error<br>(0-255: CIP error, 256-x: Library error)                           |

### 10.6.12 Reset (Reset Service)

This is a function block (FB) that executes the Reset service of a specific instance of the CIP object.

The effects of this service differ according to the CIP object.

#### ■ Icon



#### ■ Parameter

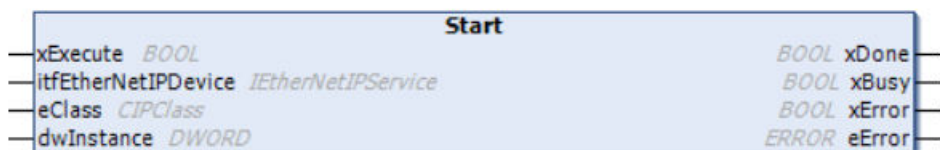
| Scope  | Name                | Type               | Description  |
|--------|---------------------|--------------------|--|
| Input  | xExecute            | BOOL               | Execution flag   |
|        | itfEtherNetIPDevice | IEtherNetIPService | EtherNet/IP device that implements the EtherNet/IP service               |
|        | eClass              | ENIP.CIPClass      | Class that executes the service  |
|        | dwInstance          | DWORD              | Instance that executes the service (0: Class level, 1-x: Instance level) |
| Output | xDone               | BOOL               | Completion flag  |
|        | xBusy               | BOOL               | Busy flag  |
|        | xError              | BOOL               | Error flag   |
|        | eError              | ENIP.ERROR         | Error (0-255: CIP error, 256-x: Library error)                           |

### 10.6.13 Start (Start Service)

This is a function block (FB) that executes the Start service of a specific instance of the CIP object.

The effects of this service differ according to the CIP object.

#### ■ Icon



## 10.6 LAN Port (EtherNet/IP)

### ■ Parameter

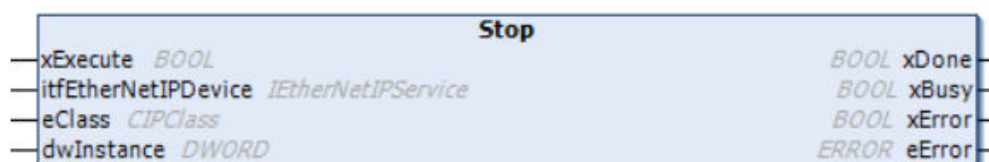
| Scope  | Name                | Type               | Description  |
|--------|---------------------|--------------------|--|
| Input  | xExecute            | BOOL               | Execution flag   |
|        | itfEtherNetIPDevice | IEtherNetIPService | EtherNet/IP device that implements the EtherNet/IP service               |
|        | eClass              | ENIP.CIPClass      | Class that executes the service  |
|        | dwInstance          | DWORD              | Instance that executes the service (0: Class level, 1-x: Instance level) |
| Output | xDone               | BOOL               | Completion flag  |
|        | xBusy               | BOOL               | Busy flag  |
|        | xError              | BOOL               | Error flag   |
|        | eError              | ENIP.ERROR         | Error (0-255: CIP error, 256-x: Library error)                           |

### 10.6.14 Stop (Stop Service)

This is a function block (FB) that executes the Stop service of a specific instance of the CIP object.

The effects of this service differ according to the CIP object.

### ■ Icon



### ■ Parameter

| Scope  | Name                | Type               | Description  |
|--------|---------------------|--------------------|--|
| Input  | xExecute            | BOOL               | Execution flag   |
|        | itfEtherNetIPDevice | IEtherNetIPService | EtherNet/IP device that implements the EtherNet/IP service               |
|        | eClass              | ENIP.CIPClass      | Class that executes the service  |
|        | dwInstance          | DWORD              | Instance that executes the service (0: Class level, 1-x: Instance level) |
| Output | xDone               | BOOL               | Completion flag  |
|        | xBusy               | BOOL               | Busy flag  |
|        | xError              | BOOL               | Error flag   |

| Scope | Name   | Type       | Description                                       |
|-------|--------|------------|---|
|       | eError | ENIP.ERROR | Error<br>(0-255: CIP error, 256-x: Library error) |

### 10.6.15 ENIP.ERROR (Message Service Instruction Error Code)

| Name                     | Value | Description  |
|--------------------------|-------|--|
| NO_ERROR                 | 0     | The service was executed normally by the specified object.   |
| CONNECTION_FAILURE       | 16#1  | The connection-related service failed due to the connection path.  |
| RESOURCE_UNAVAILABLE     | 16#2  | The object was unable to use the resources that it required to execute the requested service.  |
| INVALID_PARAM_VALUE      | 16#3  | Refer to status code 16#20 that is an appropriate value to be used in this situation.  |
| PATH_SEGMENT_ERROR       | 16#4  | The path segment identifier or segment syntax was not recognized by the processing node.<br>Path processing stops when a path segment error occurs.  |
| PATH_DESTINATION_UNKNOWN | 16#5  | The path refers to an object class, instance, or structure element that is unknown or not included in the processing node. If an unknown path destination error occurs, path processing will stop. |
| PARTIAL_TRANSFER         | 16#6  | Only part of the expected data was transferred.  |
| CONNECTION_LOST          | 16#7  | The messaging connection was lost.   |
| SERVICE_NOT_SUPPORTED    | 16#8  | The requested service is not implemented or defined for this object class or instance.   |
| INVALID_ATTRIBUTE_VALUE  | 16#9  | Invalid attribute data was detected.   |
| ATTRIBUTE_LIST_ERROR     | 16#A  | The status of the attribute of Get_Attribute_List or Set_Attribute_List response is other than zero.   |
| ALREADY_IN_REQUEST_STATE | 16#B  | The object is already in the mode or state requested by the service.   |
| OBJECT_STATE_ERROR       | 16#C  | The object cannot execute the requested service in the current mode or state.  |
| OBJECT_ALREADY_EXISTS    | 16#D  | An instance requested for the object to be created already exists.   |
| ATTRIBUTE_NOT_SETTABLE   | 16#E  | A request to change a read-only attribute was received.  |
| PRIVILEGE_VIOLATION      | 16#F  | An authority / privilege check failed.   |
| DEVICE_STATE_ERROR       | 16#10 | The current mode or state of the device prohibits the requested service from being executed.   |
| REPLY_DATA_TOO_LARGE     | 16#11 | The size of data to be sent via a response buffer is larger than the capacity of the allocated response buffer.  |
| FRAGMENTATION_OF_VALUE   | 16#12 | The service specifies an operation that fragmentates half of primitive data values which are a REAL data type.   |
| NOT_ENOUGH_DATA          | 16#13 | The service did not provide enough data to execute the specified operation.  |

## 10.6 LAN Port (EtherNet/IP)

| Name   | Value | Description   |
|--|-------|---|
| ATTRIBUTE_NOT_SUPPORTED                        | 16#14 | The attribute specified in the request is not supported.  |
| TOO_MUCH_DATA                                  | 16#15 | The service provided more data than expected.   |
| OBJECT_DOES_NOT_EXIST                          | 16#16 | The specified object does not exist in the device.  |
| SERVICE_FRAGMENTATION_SEQUENCE_NOT_IN_PROGRESS | 16#17 | The fragmentation sequence of this service is currently not active for this data.   |
| NO_STORED_ATTRIBUTE_DATA                       | 16#18 | The attribute data of this object has not been saved before the requested service is executed.  |
| STORE_OPERATION_FAILURE                        | 16#19 | The attribute data of this object has not been saved because an error occurred during the attempt to save the data.   |
| ROUTING_FAILURE_REQUEST_PACKET_TOO_LARGE       | 16#1A | The service request packet was too large to send through the network existing in the path to the destination. The routing device forcibly canceled the service.   |
| ROUTING_FAILURE_RESPONSE_PACKET_TOO_LARGE      | 16#1B | The service response packet was too large to send through the network existing in the path from the destination. The routing device forcibly canceled the service.  |
| MISSING_ATTRIBUTE_LIST_ENTRY_DATA              | 16#1C | The service did not provide attributes in the list of attributes that it requires to execute the requested operation.   |
| INVALID_ATTRIBUTE_VALUE_LIST                   | 16#1D | The service returned a list of provided attributes together with status information of invalid attributes.  |
| EMBEDDED_SERVICE_ERROR                         | 16#1E | An error occurred in the embedded service.  |
| VENDOR_SPECIFIC_ERROR                          | 16#1F | A vendor-specific error occurred. The additional code field for error response is used to define a specific error that occurred. Use this field only if the error in question does not apply to any of the error codes shown in these tables or those shown in the object class definition. |
| INVALID_PARAMETER                              | 16#20 | The parameter associated with the request is invalid. This code is used when the parameter does not meet the requirements of this specification or the requirements defined in the application object specification.  |
| WRITE_ONCE_VALUE_OR_MEDIUM_ALREADY_WRITTEN     | 16#21 | An attempt was made to write to a write-once medium (such as WORM drive or PROM) to which data has already been written or to change a value that cannot be changed once set.   |
| INVALID_REPLY_RECEIVED                         | 16#22 | An invalid response was received (for example, the response service code does not match the request service code or the response message is shorter than the expected minimum response size). This status code is useful to investigate other causes of invalid responses.                  |
| BUFFER_OVERFLOW                                | 16#23 | The size of the received message exceeds the maximum size of messages that can be handled by the receiver buffer. The entire message was discarded.   |
| MESSAGE_FORMAT_ERROR                           | 16#24 | The format of the received message is not supported by the server.  |
| KEY_FAILURE_IN_PATH                            | 16#25 | The key segment included as the first segment of the path does not match the destination module. The object-specific status indicates which part of the key check has failed.   |

## 10.6 LAN Port (EtherNet/IP)

| Name                                     | Value  | Description  |
|--|--------|--|
| PATH_SIZE_INVALID                        | 16#26  | The size of the path sent with the service request is not large enough to route the request to the object or routing data included in the path is too much.  |
| UNEXPECTED_ATTRIBUTE_IN_LIST             | 16#27  | An attempt was made to set an attribute that cannot currently be set.  |
| INVALID_MEMBER_ID                        | 16#28  | The member ID specified in the request does not exist in the specified class, instance, or attribute.  |
| MEMBER_NOT_SETTABLE                      | 16#29  | A request to change an unchangeable member was received.   |
| GROUP_2_ONLY_SERVER_GENERAL_FAILURE      | 16#2A  | This error code is issued only by DeviceNet Group 2 Only servers with 4K or less code space and is supported only instead of the server. Attributes are not supported and cannot be set.                     |
| UNKNOWN_MODBUS_ERROR                     | 16#2B  | The program for conversion from CIP to Modbus received an unknown Modbus exception code.   |
| ATTRIBUTE_NOT_GETTABLE                   | 16#2C  | A request to read an unreadable attribute was received.  |
| INSTANCE_NOT_DELETABLE                   | 16#2D  | The requested object instance cannot be deleted.   |
| SERVICE_NOT_SUPPORTED_FOR_SPECIFIED_PATH | 16#2E  | The object supports the service but does not support the specified application path (such as attributes). Note: Do not use this code for the set service. (Instead, use general status code 16#0E or 16#29.) |
| TIME_OUT                                 | 16#100 | The request has timed out.   |
| INTERFACE_MISSING                        |        | IEtherNetIPService is not implemented.   |
| REMOTE_CALL_FAILED                       |        | There is no physical connection.   |
| NULL_POINTER                             |        | A null value was entered by mistake.   |
| INVALID_DATA_SIZE                        |        | The data size is invalid.  |
| WRONG_INTERFACE_VERSION                  |        | The versions do not match. The device is not equipped with the same version of interface as the called method.   |
| NO_MEMORY                                |        | There is not enough memory.  |
| UNKNOWN_ERROR                            |        | An unknown error occurred.   |
| ABORTED                                  |        | The service was aborted.   |

### 10.6.16 ENIP.CIPClass (Service Class Code)

| Name                    | Value |
|-------------------------|-------|
| IdentityObject          | 16#1  |
| MessageRouterObject     | 16#2  |
| DeviceNetObject         | 16#3  |
| AssemblyObject          | 16#4  |
| ConnectionObject        | 16#5  |
| ConnectionManagerObject | 16#6  |
| RegisterObject          | 16#7  |

## 10.6 LAN Port (EtherNet/IP)

---

| Name                               | Value |
|------------------------------------|-------|
| DiscreteInputPointObject           | 16#8  |
| DiscreteOutputPointObject          | 16#9  |
| AnalogInputPointObject             | 16#A  |
| AnalogOutputPointObject            | 16#B  |
| PresenceSensingObject              | 16#E  |
| ParameterObject                    | 16#F  |
| ParameterGroupObject               | 16#10 |
| GroupObject                        | 16#12 |
| DiscreteInputGroupObject           | 16#1D |
| DiscreteOutputGroupObject          | 16#1E |
| DiscreteGroupObject                | 16#1F |
| AnalogInputGroupObject             | 16#20 |
| AnalogOutputGroupObject            | 16#21 |
| AnalogGroupObject                  | 16#22 |
| PositionSensorObject               | 16#23 |
| PositionControllerSupervisorObject | 16#24 |
| PositionControllerObject           | 16#25 |
| BlockSequencerObject               | 16#26 |
| CommandBlockObject                 | 16#27 |
| MotorDataObject                    | 16#28 |
| ControlSupervisorObject            | 16#29 |
| ACDCDriveObject                    | 16#2A |
| AcknowledgeHandlerObject           | 16#2B |
| OverloadObject                     | 16#2C |
| SoftstartObject                    | 16#2D |
| SelectionObject                    | 16#2E |
| S_DeviceSupervisorObject           | 16#30 |
| S_AnalogSensorObject               | 16#31 |
| S_AnalogActuatorObject             | 16#32 |
| S_SingleStageControllerObject      | 16#33 |
| S_GasCalibrationObject             | 16#34 |
| TripPointObject                    | 16#35 |
| FileObject                         | 16#37 |
| S_PartialPressureObject            | 16#38 |
| SafetySupervisorObject             | 16#39 |
| SafetyValidatorObject              | 16#3A |
| SafetyDiscreteOutputPointObject    | 16#3B |



| Name                               | Value |
|------------------------------------|-------|
| SafetyDiscreteOutputGroupObject    | 16#3C |
| SafetyDiscretelInputPointObject    | 16#3D |
| SafetyDiscretelInputGroupObject    | 16#3E |
| SafetyDualChannelOutputObject      | 16#3F |
| S_SensorCalibrationObject          | 16#40 |
| EventLogObject                     | 16#41 |
| MotionDeviceAxisObject             | 16#42 |
| TimeSyncObject                     | 16#43 |
| ModbusObject                       | 16#44 |
| OriginatorConnectionListObject     | 16#45 |
| ModbusSerialLinkObject             | 16#46 |
| DeviceLevelRingObject              | 16#47 |
| QoSObject                          | 16#48 |
| SafetyAnalogInputPointObject       | 16#49 |
| SafetyAnalogInputGroupObject       | 16#4A |
| SafetyDualChannelAnalogInputObject | 16#4B |
| SERCOSIIILinkObject                | 16#4C |
| TargetConnectionListObject         | 16#4D |
| EnergyObject                       | 16#4E |
| ElectricalEnergyObject             | 16#4F |
| Non_ElectricalEnergyObject         | 16#50 |
| BaseSwitchObject                   | 16#51 |
| SNMPObject                         | 16#52 |
| PowerManagementObject              | 16#53 |
| ControlNetObject                   | 16#F0 |
| ControlNetKeeperObject             | 16#F1 |
| ControlNetSchedulingObject         | 16#F2 |
| ConnectionConfigurationObject      | 16#F3 |
| PortObject                         | 16#F4 |
| TCPIPIInterfaceObject              | 16#F5 |
| EthernetLinkObject                 | 16#F6 |
| CompoNetLink                       | 16#F7 |
| CompoNetRepeater                   | 16#F8 |

## 10.7 SD Card Operation (File Operation)

### 10.7 SD Card Operation (File Operation)

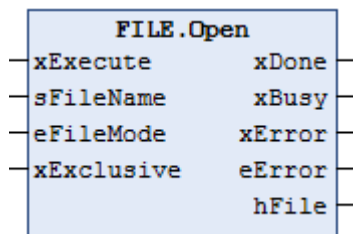
Files in the SD card inserted in the SD memory card slot can be operated.

In file operation using the GM1 Controller, WSTRING (kanji) cannot be used in the file name and directory name.

#### 10.7.1 FILE.Open (Open File)

This is a function block that opens a file or creates a new file.

##### ■ Icon



##### ■ Parameter

| Scope  | Name       | Type              | Description  |
|--------|------------|-------------------|--|
| Input  | xExecute   | BOOL              | Starts execution at the rising edge.   |
|        | sFileName  | FILE.CAA.FILENAME | Specifies the file name with an absolute path or relative path.                            |
|        | eFileMode  | FILE.MODE         | File mode  |
|        | xExclusive | BOOL              | TRUE: Exclusive access mode<br>FALSE: Multiple access mode<br>xExclusive is not supported. |
| Output | xDone      | BOOL              | TRUE: Execution is completed.  |
|        | xBusy      | BOOL              | TRUE: Execution of the FB is not completed.  |
|        | xError     | BOOL              | TRUE: An error has occurred within the FB.   |
|        | eError     | FILE.ERROR        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".                        |
|        | hFile      | FILE.CAA.HANDLE   | Handle of a file   |

##### ■ FILE.MODE (Enumeration type)

| Name   | Value | Description  |
|--------|-------|--|
| MWRITE | 0     | Overwrite mode (When the specified file does not exist, a new file is created.)    |
| MREAD  | 1     | Read mode  |
| MRDWR  | 2     | Read / write mode (When the specified file does not exist, a new file is created.) |

| Name  | Value | Description       |
|-------|-------|-------------------|
| MAPPD | 3     | Append write mode |

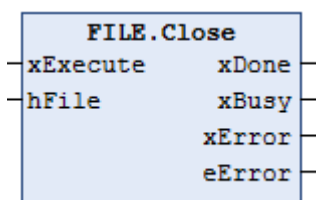
### **i** Info.

- You cannot use full size characters and the following symbols in a file name: [\], [/], [:], [\*], [?], ["], [<], [>], [[]].

### 10.7.2 FILE.Close (Close File)

This is a function block that closes a file.

#### ■ Icon



#### ■ Parameter

| Scope  | Name     | Type                 | Description  |
|--------|----------|----------------------|--|
| Input  | xExecute | BOOL                 | Starts execution at the rising edge.                                       |
|        | hFile    | FILE.CAA.H<br>HANDLE | Handle of a file to be closed<br>Specifies the handle output by FILE.Open. |
| Output | xDone    | BOOL                 | TRUE: Execution is completed.  |
|        | xBusy    | BOOL                 | TRUE: Execution of the FB is not completed.                                |
|        | xError   | BOOL                 | TRUE: An error has occurred within the FB.                                 |
|        | eError   | FILE.ERRO<br>R       | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".        |

### 10.7.3 FILE.Read (Read File)

This is a function block that reads data from the file opened.

## 10.7 SD Card Operation (File Operation)

### ■ Icon

| FILE.Read  |          |
|------------|----------|
| xExecute   | xDone    |
| xAbort     | xBusy    |
| udiTimeOut | xError   |
| hFile      | xAborted |
| pBuffer    | eError   |
| szBuffer   | szSize   |

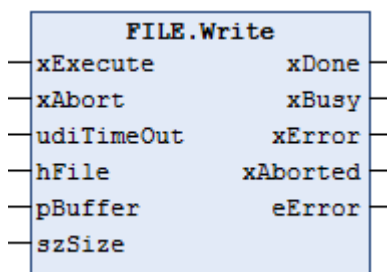
### ■ Parameter

| Scope  | Name       | Type            | Description  |
|--------|------------|-----------------|--|
| Input  | xExecute   | BOOL            | Starts execution at the rising edge.   |
|        | xAbsort    | BOOL            | TRUE: Stops execution and resets all outputs.                                |
|        | udiTimeOut | UDINT           | Timeout time until the execution is stopped (µs)                             |
|        | hFile      | FILE.CAA.HANDLE | Handle of a file<br>Specifies the handle output by FILE.Open.                |
|        | pBuffer    | FILE.CAA.PVOID  | Pointer to the data buffer to be read<br>Gets a pointer by the ADR operator. |
|        | szBuffer   | FILE.CAA.SIZE   | Size of the data buffer to be read<br>Gets a pointer by the SIZEOF operator. |
| Output | xDone      | BOOL            | TRUE: Execution is completed.  |
|        | xBusy      | BOOL            | TRUE: Execution of the FB is not completed.                                  |
|        | xError     | BOOL            | TRUE: An error has occurred within the FB.                                   |
|        | xAborted   | BOOL            | TRUE: Execution is stopped by the user.                                      |
|        | eError     | FILE.ERROR      | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".          |
|        | szSize     | FILE.CAA.SIZE   | Size of the read data buffer   |

### 10.7.4 FILE.Write (Write File)

This is a function block that writes data to the file opened.

### ■ Icon



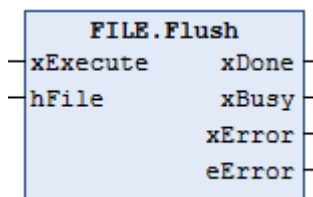
### ■ Parameter

| Scope  | Name       | Type            | Description   |
|--------|------------|-----------------|---|
| Input  | xExecute   | BOOL            | Starts execution at the rising edge.  |
|        | xAbsort    | BOOL            | TRUE: Stops execution and resets all outputs.                                   |
|        | udiTimeOut | UDINT           | Timeout time until the execution is stopped ( $\mu$ s)                          |
|        | hFile      | FILE.CAA.HANDLE | Handle of a file<br>Specifies the handle output by FILE.Open.                   |
|        | pBuffer    | FILE.CAA.PVOID  | Pointer to the data buffer to be written<br>Gets a pointer by the ADR operator. |
|        | szSize     | FILE.CAA.SIZE   | Size of the data buffer to be written<br>Gets a pointer by the SIZEOF operator. |
| Output | xDone      | BOOL            | TRUE: Execution is completed.   |
|        | xBusy      | BOOL            | TRUE: Execution of the FB is not completed.                                     |
|        | xError     | BOOL            | TRUE: An error has occurred within the FB.                                      |
|        | xAborted   | BOOL            | TRUE: Execution is stopped by the user.   |
|        | eError     | FILE.ERROR      | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".             |

### 10.7.5 FILE.Flush (Flush File)

This is a function block that flushes buffer contents to a file.

### ■ Icon



## 10.7 SD Card Operation (File Operation)

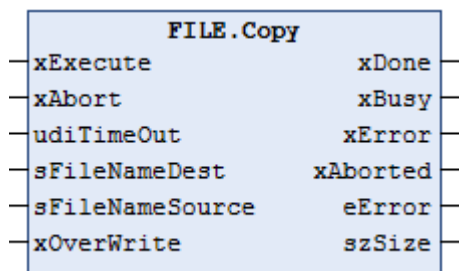
### ■ Parameter

| Scope  | Name     | Type            | Description   |
|--------|----------|-----------------|---|
| Input  | xExecute | BOOL            | Starts execution at the rising edge.                                |
|        | hFile    | FILE.CAA.HANDLE | Handle of a file<br>Specifies the handle output by FILE.Open.       |
| Output | xDone    | BOOL            | TRUE: Execution is completed.                                       |
|        | xBusy    | BOOL            | TRUE: Execution of the FB is not completed.                         |
|        | xError   | BOOL            | TRUE: An error has occurred within the FB.                          |
|        | eError   | FILE.ERROR      | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |

### 10.7.6 FILE.Copy (Copy File)

This is a function block that copies a file.

### ■ Icon



### ■ Parameter

| Scope  | Name            | Type              | Description  |
|--------|-----------------|-------------------|--|
| Input  | xExecute        | BOOL              | Starts execution at the rising edge.   |
|        | xAbort          | BOOL              | TRUE: Stops execution and resets all outputs.  |
|        | udiTimeOut      | UDINT             | Timeout time until the execution is stopped (μs)   |
|        | sFileNameDest   | FILE.CAA.FILENAME | Copy destination file name   |
|        | sFileNameSource | FILE.CAA.FILENAME | Copy source file name  |
|        | xOverWrite      | BOOL              | TRUE: Copies to overwrite an existing file.<br>FALSE: Outputs an error without copying to overwrite.<br>If FALSE is specified in a case where there is an existing file, copy is not executed. No error is output. |
| Output | xDone           | BOOL              | TRUE: Execution is completed.  |
|        | xBusy           | BOOL              | TRUE: Execution of the FB is not completed.  |
|        | xError          | BOOL              | TRUE: An error has occurred within the FB.   |

| Scope | Name     | Type          | Description   |
|-------|----------|---------------|---|
|       | xAborted | BOOL          | TRUE: Execution is stopped by the user.                             |
|       | eError   | FILE.ERROR    | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |
|       | szSize   | FILE.CAA.SIZE | Size of the copied file   |

### **i** Info.

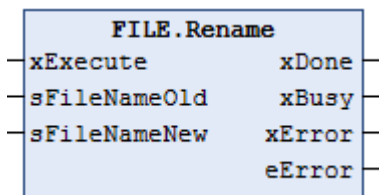
- You cannot use full size characters and the following symbols in a file name: [\], [/], [:], [\*], [?], ["], [<], [>], [[]].

## 10.7.7 FILE.Rename (Rename File)

This is a function block that changes a file name.

It is not possible to change the directory name of a directory that is currently open. Close it using the DirClose function block.

### ■ Icon



### ■ Parameter

| Scope  | Name         | Type              | Description   |
|--------|--------------|-------------------|---|
| Input  | xExecute     | BOOL              | Starts execution at the rising edge.                                |
|        | sFileNameOld | FILE.CAA.FILENAME | File name before change   |
|        | sFileNameNew | FILE.CAA.FILENAME | File name after change  |
| Output | xDone        | BOOL              | TRUE: Execution is completed.                                       |
|        | xBusy        | BOOL              | TRUE: Execution of the FB is not completed.                         |
|        | xError       | BOOL              | TRUE: An error has occurred within the FB.                          |
|        | eError       | FILE.ERROR        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |

### **i** Info.

- You cannot use full size characters and the following symbols in a file name: [\], [/], [:], [\*], [?], ["], [<], [>], [[]].

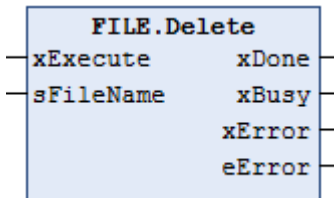
## 10.7 SD Card Operation (File Operation)

### 10.7.8 FILE.Delete (Delete File)

This is a function block that deletes a file.

It is not possible to delete a file that is currently open. Close it using the Close function block.

#### ■ Icon



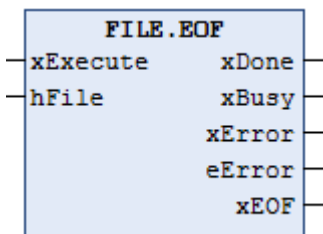
#### ■ Parameter

| Scope  | Name      | Type                  | Description   |
|--------|-----------|-----------------------|---|
| Input  | xExecute  | BOOL                  | Starts execution at the rising edge.                                |
|        | sFileName | FILE.CAA.FI<br>LENAME | File to be deleted  |
| Output | xDone     | BOOL                  | TRUE: Execution is completed.                                       |
|        | xBusy     | BOOL                  | TRUE: Execution of the FB is not completed.                         |
|        | xError    | BOOL                  | TRUE: An error has occurred within the FB.                          |
|        | eError    | FILE.ERRO<br>R        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |

### 10.7.9 FILE.EOF (End of File)

This is a function block that determines whether the current offset of a file is EOF (End Of File) or not. It can be used only when the OPEN mode is set to MREAD/MREADPLUS.

#### ■ Icon



#### ■ Parameter

| Scope | Name     | Type | Description                          |
|-------|----------|------|--------------------------------------|
| Input | xExecute | BOOL | Starts execution at the rising edge. |

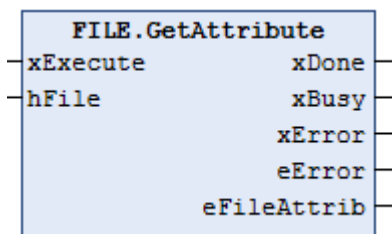


| Scope  | Name   | Type                  | Description   |
|--------|--------|-----------------------|---|
|        | hFile  | FILE.CAA.FI<br>LENAME | Handle of a file<br>Specifies the handle output by FILE.Open.       |
| Output | xDone  | BOOL                  | TRUE: Execution is completed.                                       |
|        | xBusy  | BOOL                  | TRUE: Execution of the FB is not completed.                         |
|        | xError | BOOL                  | TRUE: An error has occurred within the FB.                          |
|        | eError | FILE.ERRO<br>R        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |
|        | xEOF   | BOOL                  | File: The current offset is EOF.                                    |

### 10.7.10 FILE.GetAttribute (Get File Attribute)

This is a function block that gets file attributes.

#### ■ Icon



#### ■ Parameter

| Scope  | Name        | Type                  | Description   |
|--------|-------------|-----------------------|---|
| Input  | xExecute    | BOOL                  | Starts execution at the rising edge.                                      |
|        | hFile       | FILE.CAA.FI<br>LENAME | Handle of a file<br>Specifies the handle output by FILE.Open.             |
| Output | xDone       | BOOL                  | TRUE: Execution is completed.   |
|        | xBusy       | BOOL                  | TRUE: Execution of the FB is not completed.                               |
|        | xError      | BOOL                  | TRUE: An error has occurred within the FB.                                |
|        | eError      | FILE.ERRO<br>R        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".       |
|        | eFileAttrib | FILE.ATTRI<br>B       | TRUE: The current offset is EOF.<br>FALSE: The current offset is not EOF. |

#### ■ FILE.ATTRIB (Enumeration type)

| Name    | Value | Description  |
|---------|-------|--------------|
| ARCHIVE | 0     | Archive file |
| HIDDEN  | 1     | Hidden file  |

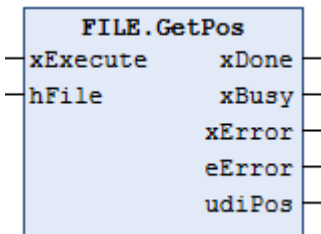
## 10.7 SD Card Operation (File Operation)

| Name     | Value | Description                       |
|----------|-------|-----------------------------------|
| NORMAL   | 2     | File without any other attributes |
| READONLY | 3     | Read only                         |

### 10.7.11 FILE.GetPos (Get File Offset)

This is a function block that gets the current offset of a file.

#### ■ Icon



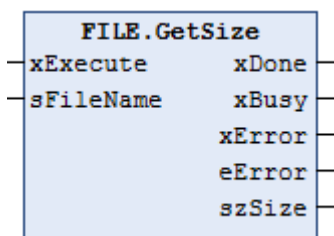
#### ■ Parameter

| Scope  | Name     | Type                  | Description   |
|--------|----------|-----------------------|---|
| Input  | xExecute | BOOL                  | Starts execution at the rising edge.                                |
|        | hFile    | FILE.CAA.FI<br>LENAME | Handle of a file<br>Specifies the handle output by FILE.Open.       |
| Output | xDone    | BOOL                  | TRUE: Execution is completed.                                       |
|        | xBusy    | BOOL                  | TRUE: Execution of the FB is not completed.                         |
|        | xError   | BOOL                  | TRUE: An error has occurred within the FB.                          |
|        | eError   | FILE.ERRO<br>R        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |
|        | udiPos   | __UXINT               | The current offset (byte) is output.                                |

### 10.7.12 FILE.GetSize (Get File Size)

This is a function block that gets the file size.

### ■ Icon



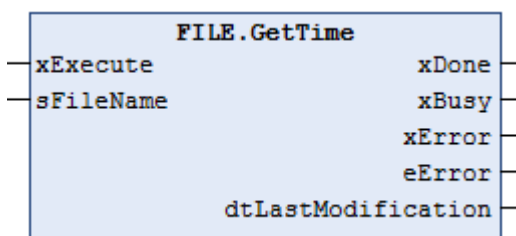
### ■ Parameter

| Scope  | Name      | Type                  | Description   |
|--------|-----------|-----------------------|---|
| Input  | xExecute  | BOOL                  | Starts execution at the rising edge.  |
|        | sFileName | FILE.CAA.FI<br>LENAME | File from which to get the file size  |
| Output | xDone     | BOOL                  | TRUE: Execution is completed.   |
|        | xBusy     | BOOL                  | TRUE: Execution of the FB is not completed.   |
|        | xError    | BOOL                  | TRUE: An error has occurred within the FB.  |
|        | eError    | FILE.ERRO<br>R        | An error ID is output.<br>Refer to " <a href="#">10.7.15 FILE.ERROR (Error ID)</a> ". |
|        | szSize    | FILE.CAA.S<br>IZE     | The file size (byte) is output.   |

## 10.7.13 FILE.GetTime (Get File Update Time)

This is a function block that gets the update time of a file.

### ■ Icon



### ■ Parameter

| Scope  | Name      | Type                  | Description                                 |
|--------|-----------|-----------------------|---|
| Input  | xExecute  | BOOL                  | Starts execution at the rising edge.        |
|        | sFileName | FILE.CAA.FI<br>LENAME | File from which to get the file update time |
| Output | xDone     | BOOL                  | TRUE: Execution is completed.               |

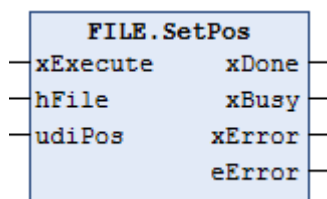
## 10.7 SD Card Operation (File Operation)

| Scope | Name               | Type          | Description  |
|-------|--------------------|---------------|--|
|       | xBusy              | BOOL          | TRUE: Execution of the FB is not completed.  |
|       | xError             | BOOL          | TRUE: An error has occurred within the FB.   |
|       | eError             | FILE.ERROR    | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".                    |
|       | dtLastModification | DATE_AND_TIME | The last update date and time is output.<br>Example: DATE_AND_TIME#2020-01-11-15:12:30 |

### 10.7.14 FILE.SetPos (Set File Offset)

This is a function block that sets the offset of a file.

#### ■ Icon



#### ■ Parameter

| Scope  | Name     | Type            | Description   |
|--------|----------|-----------------|---|
| Input  | xExecute | BOOL            | Starts execution at the rising edge.                                |
|        | hFile    | FILE.CAA.HANDLE | Handle of a file<br>Specifies the handle output by FILE.Open.       |
|        | udiPos   | __UXINT         | Offset to be set (byte)   |
| Output | xDone    | BOOL            | TRUE: Execution is completed.                                       |
|        | xBusy    | BOOL            | TRUE: Execution of the FB is not completed.                         |
|        | xError   | BOOL            | TRUE: An error has occurred within the FB.                          |
|        | eError   | FILE.ERROR      | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |

### 10.7.15 FILE.ERROR (Error ID)

This is an enumeration type error ID that is output when a function block for file operation is executed. It is used to output an error in a file operation or directory operation of the SD card.

#### ■ FILE.ERROR (Enumeration type)

| Name     | Value | Description |
|----------|-------|-------------|
| NO_ERROR | 0     | Normal end  |

## 10.7 SD Card Operation (File Operation)

---

| Name                  | Value | Description  |
|-----------------------|-------|--|
| FIRST_ERROR           | 5100  | First library specific error                                   |
| TIME_OUT              | 5101  | Timeout  |
| ABORT                 | 5102  | Aborts processing by xAbort.                                   |
| HANDLE_INVALID        | 5103  | Invalid handle   |
| NOT_EXIST             | 5104  | No file or directory exists.                                   |
| EXIST                 | 5105  | A file or directory already exists                             |
| NO_MORE_ENTRIES       | 5106  | There are no other entries.                                    |
| NOT_EMPTY             | 5107  | The file or directory is not empty.                            |
| READ_ONLY_CAA         | 5108  | The file or directory is write protected.                      |
| WRONG_PARAMETER       | 5109  | Wrong parameter  |
| ERROR_UNKNOWN         | 5110  | Unknown error  |
| WRITE_INCOMPLETE      | 5111  | Not all the data is written.                                   |
| FILE_NOT_IMPLEMENTED  | 5112  | The function is not implemented.                               |
| ASM_CREATEJOB_FAILED  | 5113  | Failed to create an AsyncManager job.                          |
| FILE_OPERATION_DENIED | 5114  | No access due to ForceFilePath / ForcelecFilePath              |
| FIRST_MF              | 5150  | First error unique to the manufacturer                         |
| LAST_ERROR            | 5199  | insert manuf. specific errors here Last library specific error |

## 10.8 SD Card Operation (Directory Operation)

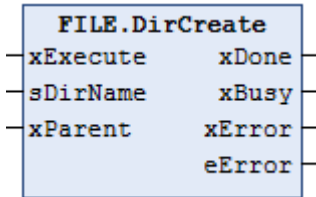
### 10.8 SD Card Operation (Directory Operation)

Directories in the SD card inserted in the SD memory card slot can be operated.

#### 10.8.1 FILE.DirCreate (Create Directory)

This is a function block that creates a directory. An error occurs if there already exists a sub-directory.

##### ■ Icon



##### ■ Parameter

| Scope  | Name     | Type                  | Description  |
|--------|----------|-----------------------|--|
| Input  | xExecute | BOOL                  | Starts execution at the rising edge.   |
|        | sDirName | FILE.CAA.FI<br>LENAME | Specifies a new directory name with an absolute path or relative path.   |
|        | xParent  | BOOL                  | TRUE: Automatically creates a non-existing sub-directory.<br>FALSE: An error occurs if there already exists a sub-directory. |
| Output | xDone    | BOOL                  | TRUE: Execution is completed.  |
|        | xBusy    | BOOL                  | TRUE: Execution of the FB is not completed.  |
|        | xError   | BOOL                  | TRUE: An error has occurred within the FB.   |
|        | eError   | FILE.ERRO<br>R        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".  |

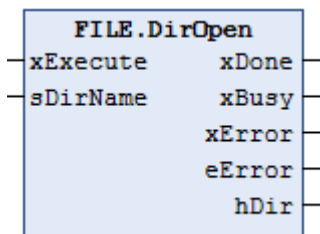
##### **i** Info.

- You cannot use full size characters and the following symbols in a directory name: [], [/], [:], [\*], [?], ["], [<], [>], [[]].

#### 10.8.2 FILE.DirOpen (Open Directory)

This is a function block that opens a directory.

### ■ Icon



### ■ Parameter

| Scope  | Name     | Type                  | Description   |
|--------|----------|-----------------------|---|
| Input  | xExecute | BOOL                  | Starts execution at the rising edge.  |
|        | sDirName | FILE.CAA.FI<br>LENAME | Specifies a directory name with an absolute path or relative path.                    |
| Output | xDone    | BOOL                  | TRUE: Execution is completed.   |
|        | xBusy    | BOOL                  | TRUE: Execution of the FB is not completed.   |
|        | xError   | BOOL                  | TRUE: An error has occurred within the FB.  |
|        | eError   | FILE.ERRO<br>R        | An error ID is output.<br>Refer to " <a href="#">10.7.15 FILE.ERROR (Error ID)</a> ". |
|        | hDir     | FILE.CAA.H<br>ANDLE   | Handle of the FILE.CAA.HANDLE directory   |

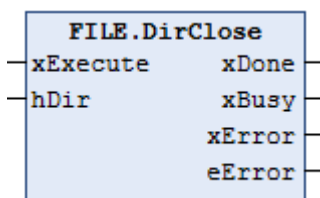
### **i** Info.

- You cannot use full size characters and the following symbols in a directory name: [ ], [ / ], [ : ], [ \* ], [ ? ], [ " ], [ < ], [ > ], [ | ].

### 10.8.3 FILE.DirClose (Close Directory)

This is a function block that closes a directory.

### ■ Icon



### ■ Parameter

| Scope | Name     | Type | Description                          |
|-------|----------|------|--------------------------------------|
| Input | xExecute | BOOL | Starts execution at the rising edge. |

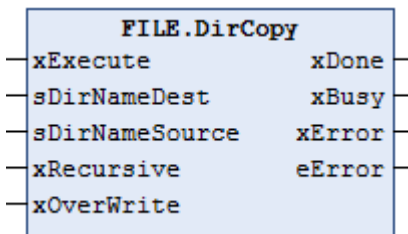
## 10.8 SD Card Operation (Directory Operation)

| Scope  | Name   | Type                  | Description   |
|--------|--------|-----------------------|---|
|        | hFile  | FILE.CAA.FI<br>LENAME | Handle of the directory to be closed<br>Specifies the handle output by FILE.Open. |
| Output | xDone  | BOOL                  | TRUE: Execution is completed.   |
|        | xBusy  | BOOL                  | TRUE: Execution of the FB is not completed.                                       |
|        | xError | BOOL                  | TRUE: An error has occurred within the FB.  |
|        | eError | FILE.ERRO<br>R        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".               |

### 10.8.4 FILE.DirCopy (Copy Directory)

This is a function block that copies a directory.

#### ■ Icon



#### ■ Parameter

| Scope  | Name           | Type                  | Description   |
|--------|----------------|-----------------------|---|
| Input  | xExecute       | BOOL                  | Starts execution at the rising edge.                                |
|        | sDirNameDest   | FILE.CAA.FI<br>LENAME | Directory name of the copy destination                              |
|        | sDirNameSource | FILE.CAA.FI<br>LENAME | Directory of the copy source  |
|        | xRecursive     | BOOL                  | TRUE: Copies the sub-directory and files.                           |
|        | xOverWrite     | BOOL                  | TRUE: Copies to overwrite an existing file.                         |
| Output | xDone          | BOOL                  | TRUE: Execution is completed.                                       |
|        | xBusy          | BOOL                  | TRUE: Execution of the FB is not completed.                         |
|        | xError         | BOOL                  | TRUE: An error has occurred within the FB.                          |
|        | eError         | FILE.ERRO<br>R        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |

#### Info.

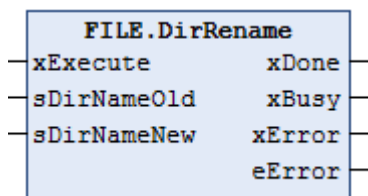
- You cannot use full size characters and the following symbols in a directory name: [], [/], [:], [\*], [?], ["], [<], [>], [[]].



### 10.8.5 FILE.DirRename (Rename Directory)

This is a function block that renames a directory name. It is not possible to change the directory name of a directory that is currently open. Close it using the DirClose function block.

#### ■ Icon



#### ■ Parameter

| Scope  | Name        | Type                 | Description   |
|--------|-------------|----------------------|---|
| Input  | xExecute    | BOOL                 | Starts execution at the rising edge.                                |
|        | sDirNameOld | FILE.CAA.FI<br>LENAM | Directory name before change  |
|        | sDirNameNew | FILE.CAA.FI<br>LENAM | Directory name after change   |
| Output | xDone       | BOOL                 | TRUE: Execution is completed.                                       |
|        | xBusy       | BOOL                 | TRUE: Execution of the FB is not completed.                         |
|        | xError      | BOOL                 | TRUE: An error has occurred within the FB.                          |
|        | eError      | FILE.ERRO<br>R       | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)". |

#### Info.

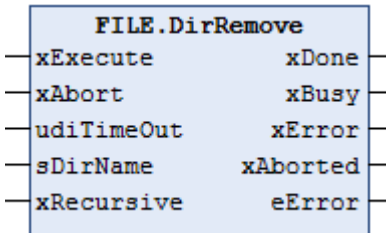
- You cannot use full size characters and the following symbols in a directory name: [ \ ], [ / ], [ : ], [ \* ], [ ? ], [ " ], [ < ], [ > ], [ | ].

### 10.8.6 FILE.DirRemove (Delete Directory)

This is a function block that deletes a directory. It is not possible to delete a directory that is currently open. Close it using the DirClose function block.

## 10.8 SD Card Operation (Directory Operation)

### ■ Icon



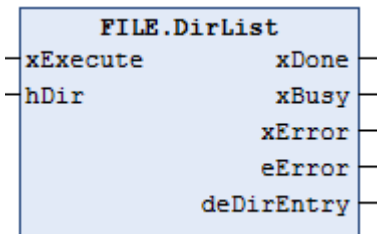
### ■ Parameter

| Scope  | Name       | Type                  | Description   |
|--------|------------|-----------------------|---|
| Input  | xExecute   | BOOL                  | Starts execution at the rising edge.  |
|        | xAbort     | BOOL                  | TRUE: Stops execution and resets all outputs.   |
|        | udiTimeOut | UDINT                 | Timeout time until the execution is stopped (µs)  |
|        | sDirName   | FILE.CAA.FI<br>LENAME | Specifies a directory name with an absolute path or relative path.  |
|        | xRecursive | BOOL                  | TRUE: Deletes the sub-directory and all files.<br>FALSE: Deletes only when the directory is empty. An error occurs if the directory is not empty. |
| Output | xDone      | BOOL                  | TRUE: Execution is completed.   |
|        | xBusy      | BOOL                  | TRUE: Execution of the FB is not completed.   |
|        | xError     | BOOL                  | TRUE: An error has occurred within the FB.  |
|        | xAborted   | BOOL                  | TRUE: Execution is stopped by the user.   |
|        | eError     | FILE.ERRO<br>R        | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".   |

### 10.8.7 FILE.DirList (Directory List)

This is a function block that outputs a list of directories and files inside the directory.

### ■ Icon



### ■ Parameter

| Scope  | Name       | Type                | Description  |
|--------|------------|---------------------|--|
| Input  | xExecute   | BOOL                | Starts execution at the rising edge.   |
|        | hDir       | FILE.CAA.HANDLE     | Directory from which to output a list<br>Specifies the handle output by FILE.Open. |
| Output | xDone      | BOOL                | TRUE: Execution is completed.  |
|        | xBusy      | BOOL                | TRUE: Execution of the FB is not completed.  |
|        | xError     | BOOL                | TRUE: An error has occurred within the FB.   |
|        | eError     | FILE.ERROR          | An error ID is output.<br>Refer to "10.7.15 FILE.ERROR (Error ID)".                |
|        | deDirEntry | FILE.FILE_DIR_ENTRY | Files and directories are output.  |

### ■ FILE\_DIR\_ENTRY (Structure)

| Member             | Type              | Description  |
|--------------------|-------------------|--|
| sEntry             | FILE.CAA.FILENAME | Directory or file name                                     |
| szSize             | FILE.CAA.SIZE     | File size  |
| xDirectory         | BOOL              | TRUE: Directory<br>FALSE: File                             |
| xExclusive         | BOOL              | TRUE: Exclusive access mode<br>FALSE: Multiple access mode |
| dtLastModification | DATE_AND_TIME     | Last update date and time.                                 |

## 10.9 Clock Setting

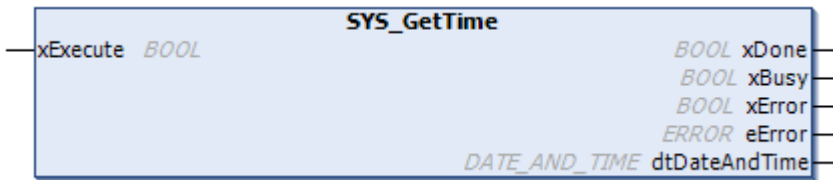
### 10.9 Clock Setting

This section describes function blocks that are used to set the clock of the GM1 Controller. Enter a function block name by using the RTCLK (namespace).

#### 10.9.1 SYS\_GetTime (Get Time)

This is a function block (FB) that gets the current local time

##### ■ Icon



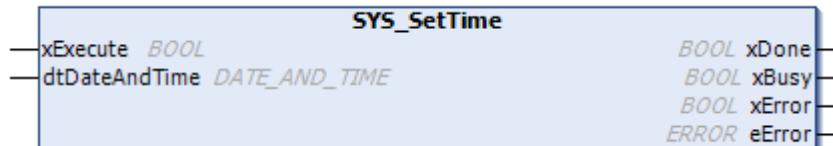
##### ■ Parameter

| Scope  | Name          | Type  | Description                                 |
|--------|---------------|-------|---|
| Input  | xExecute      | BOOL  | TRUE: Active<br>FALSE: Stop                 |
| Output | xDone         | BOOL  | TRUE: The function block is normally ended. |
|        | xBusy         | BOOL  | TRUE: The function block is active.         |
|        | xError        | BOOL  | TRUE: An error has occurred.                |
|        | eError        | ERROR | Details of error contents                   |
|        | dtDateAndTime | DT    | Current local time                          |

#### 10.9.2 SYS\_SetTime (Set Time)

This is a function block (FB) that sets the current local time.

##### ■ Icon



##### ■ Parameter

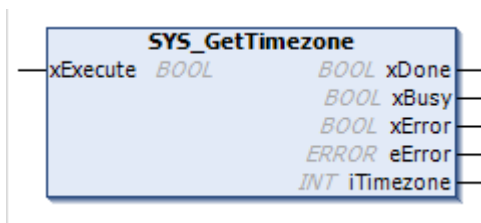
| Scope | Name     | Type | Description  |
|-------|----------|------|--------------|
| Input | xExecute | BOOL | TRUE: Active |

| Scope  | Name          | Type  | Description                                 |
|--------|---------------|-------|---|
|        |               |       | FALSE: Stop                                 |
|        | dtDateAndTime | DT    | Current time to be set                      |
| Output | xDone         | BOOL  | TRUE: The function block is normally ended. |
|        | xBusy         | BOOL  | TRUE: The function block is active.         |
|        | xError        | BOOL  | TRUE: An error has occurred.                |
|        | eError        | ERROR | Details of error contents                   |

### 10.9.3 SYS\_GetTimezone (Get Time Zone Information)

This is a function block (FB) that gets the time zone information.

#### ■ Icon



#### ■ Parameter

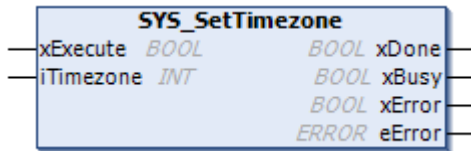
| Scope  | Name      | Type  | Description                                 |
|--------|-----------|-------|---|
| Input  | xExecute  | BOOL  | TRUE: Active<br>FALSE: Stop                 |
| Output | xDone     | BOOL  | TRUE: The function block is normally ended. |
|        | xBusy     | BOOL  | TRUE: The function block is active.         |
|        | xError    | BOOL  | TRUE: An error has occurred.                |
|        | eError    | ERROR | Details of error contents                   |
|        | iTimezone | INT   | Time zone information (Offset from UTC)     |

### 10.9.4 SYS\_SetTimezone (Set Time Zone Information)

This is a function block (FB) that sets the time zone information.

## 10.9 Clock Setting

### ■ Icon



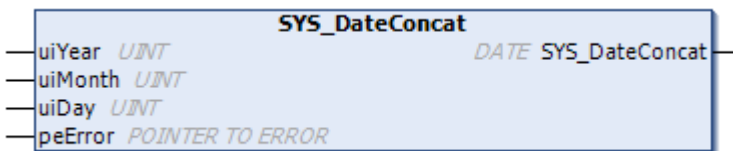
### ■ Parameter

| Scope  | Name      | Type  | Description                                 |
|--------|-----------|-------|---|
| Input  | xExecute  | BOOL  | TRUE: Active<br>FALSE: Stop                 |
|        | iTimezone | INT   | Time zone information (Offset from UTC)     |
| Output | xDone     | BOOL  | TRUE: The function block is normally ended. |
|        | xBusy     | BOOL  | TRUE: The function block is active.         |
|        | xError    | BOOL  | TRUE: An error has occurred.                |
|        | eError    | ERROR | Details of error contents                   |

### 10.9.5 SYS\_DateConcat (Convert from UINT Type to DATE Type)

This is a function (FUN) that converts a UINT type date to a DATE type.

### ■ Icon



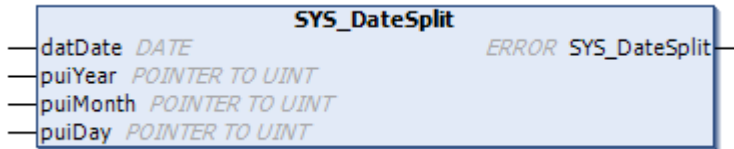
### ■ Parameter

| Scope  | Name           | Type             | Description  |
|--------|----------------|------------------|--|
| Input  | uiYear         | UINT             | Year: 1970 to 2099   |
|        | uiMonth        | UINT             | Month: 1 to 12   |
|        | uiDay          | UINT             | Day: 1 to 31   |
|        | peError        | POINTER TO ERROR | Pointer to the error information storage location                  |
| Output | SYS_DateConcat | DATE             | Return value: Returns DT#1970-01-01 if the input value is invalid. |

### 10.9.6 SYS\_DateSplit (Convert from DATE Type to UINT Type)

This is a function (FUN) that converts a DATE type date to a UINT type.

#### ■ Icon



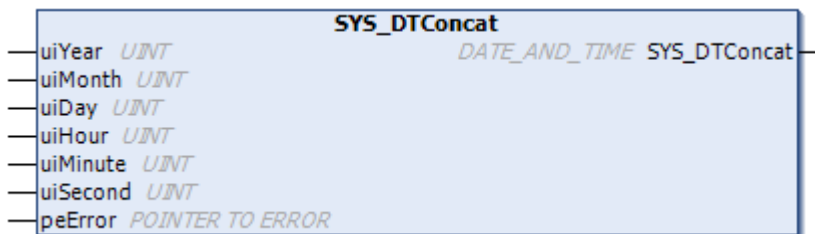
#### ■ Parameter

| Scope  | Name          | Type            | Description   |
|--------|---------------|-----------------|---|
| Input  | datDate       | DATE            | Date data   |
|        | puiYear       | POINTER TO UINT | Pointer to the year data storage location: 1970 to 2099 |
|        | puiMonth      | POINTER TO UINT | Pointer to the month data storage location: 1 to 12     |
|        | puiDay        | POINTER TO UINT | Pointer to the day data storage location: 1 to 31       |
| Output | SYS_DateSplit | ERROR           | Return value: Error information                         |

### 10.9.7 SYS\_DTConcat (Convert from UINT Type to DT Type)

This is a function (FUN) that converts a UINT type date and time to a DT type.

#### ■ Icon



#### ■ Parameter

| Type  | Name    | Type | Description        |
|-------|---------|------|--------------------|
| Input | uiYear  | UINT | Year: 1970 to 2099 |
|       | uiMonth | UINT | Month: 1 to 12     |
|       | uiDay   | UINT | Day: 1 to 31       |

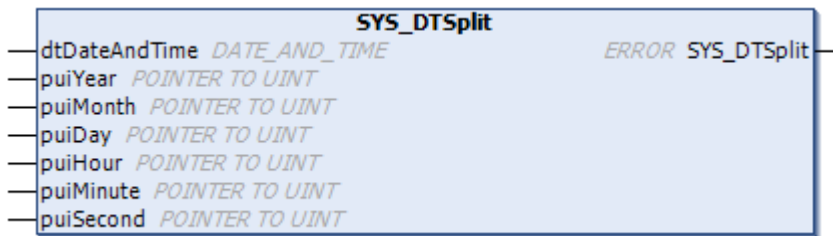
## 10.9 Clock Setting

| Type   | Name         | Type             | Description   |
|--------|--------------|------------------|---|
|        | uiHour       | UINT             | Hour: 0 to 23   |
|        | uiMinute     | UINT             | Minute: 0 to 59   |
|        | uiSecond     | UINT             | Second: 0 to 59   |
|        | peError      | POINTER TO ERROR | Pointer to the error information  |
| Output | SYS_DTConcat | DT               | Return value: Returns DT#1970-01-01-00:00:00 if the input value is invalid. |

### 10.9.8 SYS\_DTSplit (Convert from UINT Type to DT Type)

This is a function (FUN) that converts a UINT type date and time to a DT type.

#### ■ Icon



#### ■ Parameter

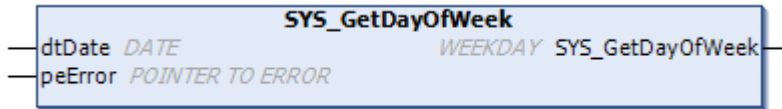
| Scope  | Name          | Type            | Description   |
|--------|---------------|-----------------|---|
| Input  | dtDateAndTime | DT              | Date and time data                                      |
|        | uiYear        | POINTER TO UINT | Pointer to the year data storage location: 1970 to 2099 |
|        | uiMonth       | POINTER TO UINT | Pointer to the month data storage location: 1 to 12     |
|        | uiDay         | POINTER TO UINT | Pointer to the day data storage location: 1 to 31       |
|        | uiHour        | POINTER TO UINT | Pointer to the hour data storage location: 0 to 23      |
|        | uiMinute      | POINTER TO UINT | Pointer to the minute data storage location: 0 to 59    |
|        | uiSecond      | POINTER TO UINT | Pointer to the second data storage location: 0 to 59    |
| Output | SYS_DTsplit   | ERROR           | Return value: Error information                         |



**10.9.9 SYS\_GetDayOfWeek (Get Day of the Week)**

This is a function (FUN) that gets the day of the week from the DATE type date.

■ **Icon**



■ **Parameter**

| Scope  | Name             | Type             | Description                      |
|--------|------------------|------------------|----------------------------------|
| Input  | dtDate           | DATE             | Date data                        |
|        | peError          | POINTER TO ERROR | Pointer to the error information |
| Output | SYS_GetDayOfWeek | RTCLK.WEEKDAY    | Return value: Day of the week    |

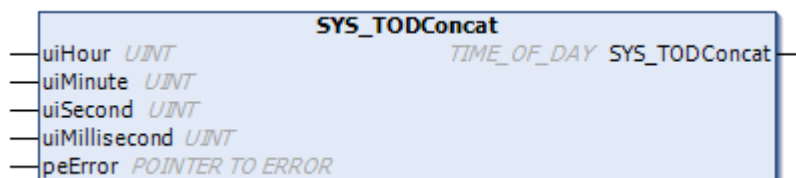
**RTCLK.WEEKDAY (Day of the week)**

| Name      | Value | Description |
|-----------|-------|-------------|
| SUNDAY    | 16#00 | Sunday      |
| MONDAY    | 16#01 | Monday      |
| TUESDAY   | 16#02 | Tuesday     |
| WEDNESDAY | 16#03 | Wednesday   |
| THURSDAY  | 16#04 | Thursday    |
| FRIDAY    | 16#05 | Friday      |
| SATURDAY  | 16#06 | Saturday    |

**10.9.10 SYS\_TODConcat (Convert from UINT Type to TOD Type)**

This is a function (FUN) that converts a UINT type time with milliseconds to a TOD type.

■ **Icon**



## 10.9 Clock Setting

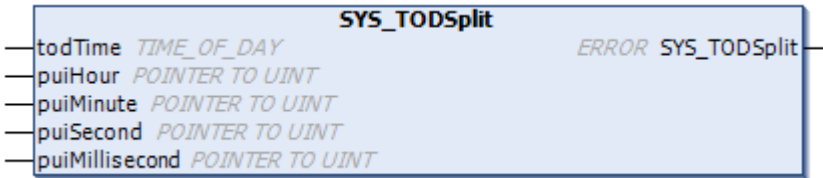
### ■ Parameter

| Scope  | Name          | Type             | Description   |
|--------|---------------|------------------|---|
| Input  | uiHour        | UINT             | Hour: 0 to 23   |
|        | uiMinute      | UINT             | Minute: 0 to 59   |
|        | uiSecond      | UINT             | Second: 0 to 59   |
|        | uiMillisecond | UINT             | Millisecond: 0 to 999   |
|        | peError       | POINTER TO ERROR | Pointer to the error information                                    |
| Output | SYS_TODConcat | TOD              | Return value<br>Returns TOD#00:00:00 if the input value is invalid. |

### 10.9.11 SYS\_TODSplit (Convert from TOD Type to UINT Type)

This is a function (FUN) that converts a TOD type time with milliseconds to a UINT type.

### ■ Icon



### ■ Parameter

| Scope  | Name           | Type            | Description  |
|--------|----------------|-----------------|--|
| Input  | todTime        | TOD             | Time (hour, minute, second) with millisecond data          |
|        | puiHour        | POINTER TO UINT | Pointer to the hour data storage location: 0 to 23         |
|        | puiMinute      | POINTER TO UINT | Pointer to the minute data storage location: 0 to 59       |
|        | puiSecond      | POINTER TO UINT | Pointer to the second data storage location: 0 to 59       |
|        | puiMillisecond | POINTER TO UINT | Pointer to the millisecond data storage location: 0 to 999 |
| Output | SYS_TODSplit   | ERROR           | Return value: Error information                            |

**10.9.12 ERROR (Clock Instruction Error Code)**

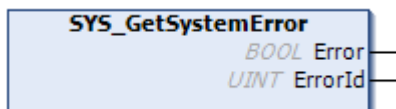
| Name                | Value | Description   |
|---------------------|-------|---|
| NO_ERROR            | 0     | No error  |
| FIRST_ERROR         | 5700  | First error unique to the library                   |
| TIME_OUT            | 5751  | The time limit is exceeded.                         |
| NOT_AVAILABLE       | 5752  | Not available.                                      |
| INPUT_VALID         | 5753  | Invalid input value                                 |
| DTU_ERROR_UNKNOWN   | 5754  | Unknown error                                       |
| DTU_WRONG_PARAMETER | 5755  | Wrong parameter                                     |
| DTU_TZI_NOT_SET     | 5756  | The time zone information has not been initialized. |
| FIRST_MF            | 5770  | First error unique to the manufacturer              |
| LAST_ERROR          | 5799  | Last error unique to the library                    |

### 10.10 System Data

#### 10.10.1 SYS\_GetSystemError (Get System Error)

This is a function block that gets the information of a system error that has occurred in the GM1 Controller.

##### ■ Icon



##### ■ Parameter

| Scope  | Name    | Type  | Description                             |
|--------|---------|-------|---|
| Input  | None    |       |   |
| Output | Error   | BOOL  | TRUE: An error has occurred.            |
|        | ErrorId | UDINT | Error ID of the error that has occurred |

##### Info.

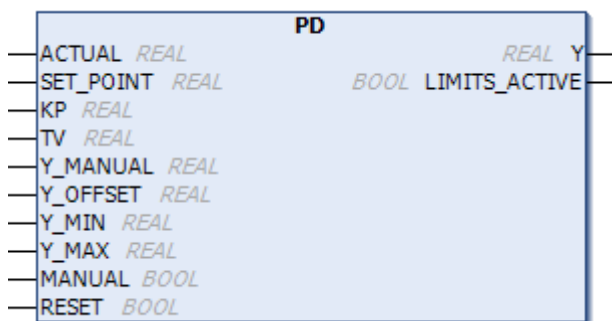
- For the error IDs, refer to the *GM1 Controller RTEX User's Manual (Operation Edition)* or *GM1 Controller EtherCAT User's Manual (Operation Edition)*.

## 10.11 PID Control

### 10.11.1 PD (PD Control)

This is a function block (FB) that performs PD control. P control can be performed when TV is set to 0.

#### ■ Icon



#### ■ Parameter

| Scope  | Name          | Type | Default | Description  |
|--------|---------------|------|---------|--|
| Input  | ACTUAL        | REAL | -       | Current value  |
|        | SET POINT     | REAL | -       | Target value   |
|        | KP            | REAL | -       | Proportionality constant P   |
|        | TV            | REAL | -       | Derivative time D (unit: s)  |
|        | Y_MANUAL      | REAL | -       | A value output to output value (Y) when MANUAL = TRUE is set.  |
|        | Y_OFFSET      | REAL | -       | An offset value of output value (Y)  |
|        | Y_MIN         | REAL | -       | A lower limit value of output value (Y)  |
|        | Y_MAX         | REAL | -       | An upper limit value of output value (Y)   |
|        | MANUAL        | BOOL | -       | TRUE: The value set in Y_MANUAL is output to output value (Y).   |
|        | RESET         | BOOL | -       | TRUE: Output value (Y) is reset. Output value (Y) is set to the offset value (Y_OFFSET) to reset the integral portion. |
| Output | Y             | REAL | -       | Output value   |
|        | LIMITS_ACTIVE | BOOL | FALSE   | TRUE: Output value (Y) is outside the range defined by Y_MIN/Y_MAX.  |

## 10.11 PID Control

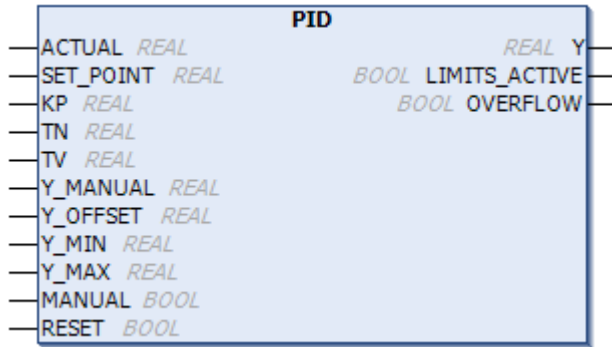
### 10.11.2 PID (PID Control)

This is a function block (FB) that performs PID control. Cycle time is automatically measured and PID operation is executed. PI control can be performed when TV is set to 0.

#### **i** Info.

- Cycle time is a time passed while the FB is called twice.

#### ■ Icon



#### ■ Parameter

| Scope  | Name          | Type | Default  | Description   |
|--------|---------------|------|--|---|
| Input  | ACTUAL        | REAL | -  | Current value   |
|        | SET POINT     | REAL | -  | Target value  |
|        | KP            | REAL | -  | Proportionality constant P  |
|        | TN            | REAL | -  | Integral time I (unit: s)   |
|        | TV            | REAL | -  | Derivative time D (unit: s)   |
|        | Y_MANUAL      | REAL | -  | A value output to output value (Y) when MANUAL = TRUE is set.       |
|        | Y_OFFSET      | REAL | -  | An offset value of output value (Y)                                 |
|        | Y_MIN         | REAL | -  | A lower limit value of output value (Y)                             |
|        | Y_MAX         | REAL | -  | An upper limit value of output value (Y)                            |
|        | MANUAL        | BOOL | -  | TRUE: The value set in Y_MANUAL is output to output value (Y).      |
| RESET  | BOOL          | -    | TRUE: Output value (Y) is reset. Output value (Y) is set to the offset value (Y_OFFSET) to reset the integral portion. |   |
| Output | Y             | REAL | -  | Output value  |
|        | LIMITS_ACTIVE | BOOL | FALSE  | TRUE: Output value (Y) is outside the range defined by Y_MIN/Y_MAX. |

| Scope | Name     | Type | Default | Description                    |
|-------|----------|------|---------|--------------------------------|
|       | OVERFLOW | BOOL | -       | TRUE: Integer portion overflow |

### **i** Info.

- The maximum preciseness is 1 ms and thus the precision decreases when the operation is executed in a short cycle time.  
For instance, if the cycle time is 1 ms, PID control measures the cycle time as 2 ms or 0 ms in some cases.  
If the operation is executed in a short cycle time, use of PID\_FIXCYCLE is recommended.

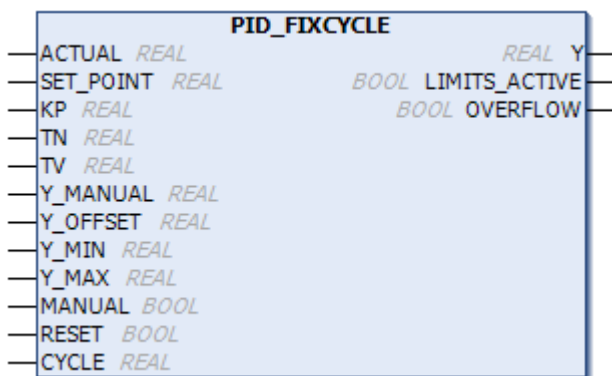
### 10.11.3 PID\_FIXCYCLE [PID Control (Any Cycle Time)]

This is a function block (FB) that performs PID control. Cycle time can be manually set. PID operation is executed over the set cycle time. Except for the manually set cycle time, this FB operates in the same way as PID function blocks.

### **i** Info.

- Cycle time is a time passed while the FB is called twice.

#### ■ Icon



#### ■ Parameter

| Scope | Name      | Type | Default | Description                 |
|-------|-----------|------|---------|-----------------------------|
| Input | ACTUAL    | REAL | -       | Current value               |
|       | SET POINT | REAL | -       | Target value                |
|       | KP        | REAL | -       | Proportionality constant P  |
|       | TN        | REAL | -       | Integral time I (unit: s)   |
|       | TV        | REAL | -       | Derivative time D (unit: s) |

## 10.11 PID Control

| Scope  | Name          | Type | Default | Description   |
|--------|---------------|------|---------|---|
|        | Y_MANUAL      | REAL | -       | A value output to output value (Y) when MANUAL = TRUE is set.   |
|        | Y_OFFSET      | REAL | -       | An offset value of output value (Y)   |
|        | Y_MIN         | REAL | -       | A lower limit value of output value (Y)   |
|        | Y_MAX         | REAL | -       | An upper limit value of output value (Y)  |
|        | MANUAL        | BOOL | -       | TRUE: The value set in Y_MANUAL is output to output value (Y).  |
|        | RESET         | BOOL | -       | TRUE: Output value (Y) is reset.<br>Output value (Y) is set to the offset value (Y_OFFSET) to reset the integral portion. |
|        | CYCLE         | REAL | -       | A time passed while the FB is called twice  |
| Output | Y             | REAL | -       | Output value  |
|        | LIMITS_ACTIVE | BOOL | FALSE   | TRUE: Output value (Y) is outside the range defined by Y_MIN/Y_MAX.   |
|        | OVERFLOW      | BOOL | -       | TRUE: Integer portion overflow  |



# 11 Function Blocks for the Pulse Output Unit

---

|  |       |
|--|-------|
| 11.1 Basic Configuration of Function Blocks for the Pulse Output Unit .... | 11-2  |
| 11.1.1 Specifications of the Function Block.....                           | 11-2  |
| 11.1.2 Notes for Executing the Function Block .....                        | 11-3  |
| 11.2 Function Blocks for the Pulse Output Unit .....                       | 11-4  |
| 11.2.1 PG_Power .....  | 11-4  |
| 11.2.2 PG_Jog .....  | 11-5  |
| 11.2.3 PG_MoveAbsolute .....   | 11-6  |
| 11.2.4 PG_MoveRelative .....   | 11-7  |
| 11.2.5 PG_LatchPosition.....   | 11-8  |
| 11.2.6 PG_Pulser .....   | 11-10 |
| 11.2.7 PG_Stop.....  | 11-11 |
| 11.2.8 PG_Home.....  | 11-12 |
| 11.2.9 PG_SetPosition .....  | 11-13 |
| 11.2.10 PG_WriteParameter .....  | 11-14 |
| 11.2.11 PG_ReadParameter .....   | 11-17 |
| 11.2.12 PG_ClearError.....   | 11-18 |
| 11.2.13 PG_ReadStatus .....  | 11-18 |
| 11.3 Error Codes.....  | 11-21 |
| 11.3.1 Error Check Method .....  | 11-21 |
| 11.3.2 PG_ERROR .....  | 11-22 |

## 11.1 Basic Configuration of Function Blocks for the Pulse Output Unit

### 11.1 Basic Configuration of Function Blocks for the Pulse Output Unit

This section describes the basic configuration of the function block.

#### 11.1.1 Specifications of the Function Block

##### ■ Common parameters

The following table lists common parameters used by the function blocks for the GM1 Positioning Unit.

| Scope  | Parameter      | Description   |
|--------|----------------|---|
| Input  | UnitID         | Specify the connection order for positioning units where function blocks are applied in a range from 1 to 15.   |
|        | AxisNo         | Specify the No. of the axis where function blocks are applied in a range from 1 to 4.   |
|        | Execute        | This is a trigger that executes the function block. <ul style="list-style-type: none"><li>It executes the function block at the rising edge when the Execute input becomes TRUE.</li><li>When the Execute input is set to FALSE after a function block is processed, one of the following parameters is cleared: CommandAborted, Done, or Error.</li><li>Even when the Execute input is set to FALSE while a function block is executed (Busy = TRUE), the function block keeps operating.<br/>At this time, when operation is finished, CommandAborted, Done or Error is held for one cycle.</li></ul> |
|        | Enable         | This enables an execution of a function block. <ul style="list-style-type: none"><li>It executes the function block at the rising edge when the Enable input becomes TRUE.</li><li>There are some parameter that can be changed even when the Enable input is set to TRUE. Check this for every function block.</li><li>When the Enable input is set to FALSE, the function block processing is stopped.</li></ul>  |
| Output | Busy           | The Busy output is set to TRUE while the function block is being executed. <sup>(Note 1)</sup>  |
|        | Done           | The Done output is set to TRUE when the function block processing is completed. <sup>(Note 1)</sup>   |
|        | CommandAborted | When the function block processing is interrupted, the CommandAborted output is set to TRUE. <sup>(Note 1)</sup>  |
|        | Error          | When an error occurs while the function block is executed, the Error output is set to TRUE.   |
|        | ErrorID        | Error information can be checked when an error occurs.  |
|        | Valid          | When an output becomes valid, this output becomes TRUE.   |

(Note 1) After a function block is processed, one of the following parameters is set to TRUE: CommandAborted, Done, or Error.

##### ■ Tasks

Either MotionTask or UserTask can be executed.

Use the same task for performing a process for the same axis.

### ■ Simulation mode

The function block is not supported in the simulation mode. If a function block is executed in the simulation mode, the PG\_NOT\_SUPPORTED error is issued.

### 11.1.2 Notes for Executing the Function Block

#### ■ Busy state

If the function block is busy (Busy = TRUE), call the function block at every cycle when executing a task.

#### Info.

- Exceptions, if any, will be described in the specifications for each function block.

#### ■ Interruption of function block processing

- The following function blocks cannot be executed at the same time on the same axis: PG\_Jog, PG\_MoveRelative, PG\_MoveAbsolute, PG\_LatchPosition, PG\_Home, and PG\_Pulser

The function block executed first takes precedence over the others.

As for the function block that is executed later, CommandAborted is set to TRUE and processing is not started.

- PG\_Stop takes precedence over any other control.  
If operation is stopped by executing PG\_Stop, CommandAborted is set to TRUE for all other functions blocks and their controls are stopped
- If a stop operation (RUN → STOP) is applied in the middle of executing the function block, the pulse output unit stops outputting pulse signals.  
When operation is resumed, CommandAborted is set to TRUE for the function block being operated and its processing is interrupted.
- If an error occurs in a pulse output unit, the pulse output unit stops outputting pulse signals.  
For the function block that is being executed, the PG\_AXIS\_UNIT\_ERROR is issued and its processing is interrupted.

#### ■ Continuous execution of a function block

In a cycle where the Done output is set to TRUE, re-execute the function block (Execute = TRUE) to continue execution of the function block.

## 11.2 Function Blocks for the Pulse Output Unit

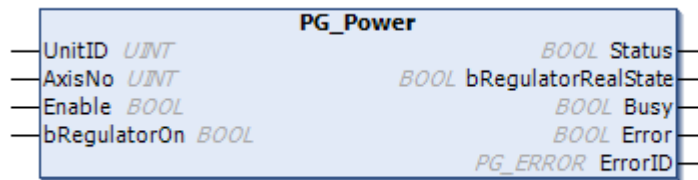
### 11.2 Function Blocks for the Pulse Output Unit

Various motion operations can be realized by using the function blocks for the pulse output unit. This section describes motion control function blocks for the single axis.

#### 11.2.1 PG\_Power

This is a function block (FB) that performs servo ON/OFF control on the axis of the pulse output unit.

##### ■ Icon



##### ■ Parameter

| Scope  | Parameter name      | Type     | Default  | Description  |
|--------|---------------------|----------|----------|--|
| Input  | UnitID              | UINT     | -        | Specifies the unit ID.   |
|        | AxisNo              | UINT     | -        | Specifies the axis No.   |
|        | Enable              | BOOL     | FALSE    | TRUE: Execution of the FB is enabled.                                  |
|        | bRegulatorOn        | BOOL     | FALSE    | TRUE: Servo ON<br>FALSE: Servo OFF                                     |
| Output | Status              | BOOL     | FALSE    | TRUE: The axis can be executed.<br>FALSE: The axis cannot be executed. |
|        | bRegulatorRealState | BOOL     | FALSE    | TRUE: Servo ON state<br>FALSE: Servo OFF state                         |
|        | Busy                | BOOL     | FALSE    | TRUE: FB is in progress.   |
|        | Error               | BOOL     | FALSE    | TRUE: FB is abnormally completed.                                      |
|        | ErrorID             | PG_ERROR | NO_ERROR | An error ID is output.   |

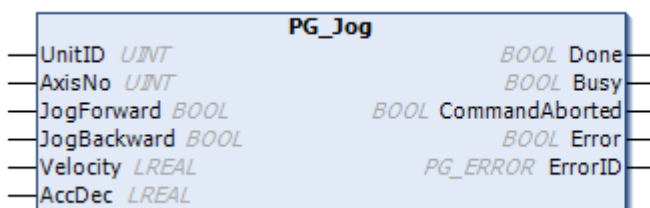
##### ■ Notes for Executing the Function Block

- After executing PG\_Power, check the servo ON state using the bRegulatorRealState output parameter.
- It is not necessary to call PG\_Power at every cycle.

### 11.2.2 PG\_Jog

This is a function block (FB) that causes the axis of the pulse output unit to keep traveling in a forward or backward direction.

#### ■ Icon



#### ■ Parameter

| Scope  | Parameter name | Type     | Default  | Description  |
|--------|----------------|----------|----------|--|
| Input  | UnitID         | UINT     | -        | Specifies the unit ID.   |
|        | AxisNo         | UINT     | -        | Specifies the axis No.   |
|        | JogForward     | BOOL     | FALSE    | TRUE: Travels in a forward direction.                          |
|        | JogBackward    | BOOL     | FALSE    | TRUE: Travels in a backward direction.                         |
|        | Velocity       | LREAL    | 0        | Specifies the target speed (u/s).                              |
|        | AccDec         | LREAL    | 0        | Specifies the acceleration / deceleration (u/s <sup>2</sup> ). |
| Output | Done           | BOOL     | FALSE    | TRUE: FB operation is completed.                               |
|        | Busy           | BOOL     | FALSE    | TRUE: FB operation is in progress.                             |
|        | CommandAborted | BOOL     | FALSE    | TRUE: FB operation is interrupted.                             |
|        | Error          | BOOL     | FALSE    | TRUE: FB is abnormally completed.                              |
|        | ErrorID        | PG_ERROR | NO_ERROR | An error ID is output.   |

#### ■ Operations when the function block is executed

- Set either JogForward or JogBackward to TRUE to execute the function block.
- Depending on the direction in which the axis is to be moved, set either JogForward or JogBackward to TRUE
- If JogForward and JogBackward are simultaneously executed, the PG\_JOG\_INVALID\_REQUEST error is issued.



- When executing again after the occurrence of the error, set both JogForward and JogBackward to FALSE once.

- When JogForward is switched to JogBackward (or JogBackward is switched to JogForward), the operation is switched only after the pulse output unit completes outputting pulse output currently controlled.

## 11.2 Function Blocks for the Pulse Output Unit

- The Velocity input can be changed in the middle of operation (in a situation where either JogForward or JogBackward is set to TRUE).
- The AccDec input cannot be changed during operation.

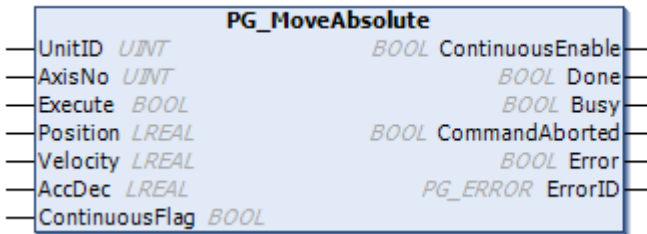


- When executing the function block, use the PG\_Power function block in advance to set to the servo ON state.
- If the function block is executed in the servo OFF state, the PG\_AXIS\_NOT\_READY\_FOR\_MOTION error is issued.

### 11.2.3 PG\_MoveAbsolute

This is a function block (FB) that causes the axis of the pulse output unit to travel to a position specified as an absolute position.

#### ■ Icon



#### ■ Parameter

| Scope  | Parameter name   | Type     | Default  | Description  |
|--------|------------------|----------|----------|--|
| Input  | UnitID           | UINT     | -        | Specifies the unit ID.   |
|        | AxisNo           | UINT     | -        | Specifies the axis No.   |
|        | Execute          | BOOL     | FALSE    | Starts execution at the rising edge.                           |
|        | Position         | LREAL    | 0        | Specifies the target position (u).                             |
|        | Velocity         | LREAL    | 0        | Specifies the maximum velocity (u/s).                          |
|        | AccDec           | LREAL    | 0        | Specifies the acceleration / deceleration (u/s <sup>2</sup> ). |
|        | ContinuousFlag   | BOOL     | FALSE    | FALSE: E-point control<br>TRUE: P-point control                |
| Output | ContinuousEnable | BOOL     | FALSE    | TRUE: Position, Velocity, and AccDec can be changed.           |
|        | Done             | BOOL     | FALSE    | TRUE: FB operation is completed.                               |
|        | Busy             | BOOL     | FALSE    | TRUE: FB operation is in progress.                             |
|        | CommandAborted   | BOOL     | FALSE    | TRUE: FB operation is interrupted.                             |
|        | Error            | BOOL     | FALSE    | TRUE: FB is abnormally completed.                              |
|        | ErrorID          | PG_ERROR | NO_ERROR | An error ID is output.   |

### ■ Operations when the function block is executed

- Absolute value control is performed according to the position specified in the Position input. (Coordinates: -2147483648 to 2147483647u)
- To use the E-point control, specify as follows.
  - Specifies ContinuousFlag = FALSE and execute.
- To use the P-point control, specify as follows.
  - Specifies ContinuousFlag = TRUE and execute.
  - When the ContinuousEnable output becomes TRUE after PG\_MoveAbsolute is started, the Position, Velocity, and AccDec inputs can be rewritten. Change the inputs using the rising edge when the Execute input becomes TRUE as a trigger.
  - If rewriting is performed when the ContinuousEnable output is FALSE, the PG\_FB\_IN\_BUSY error is issued.
  - If rewriting is not executed, the execution of PG\_MoveAbsolute is completed (Done=TRUE) when the pulse output is completed.

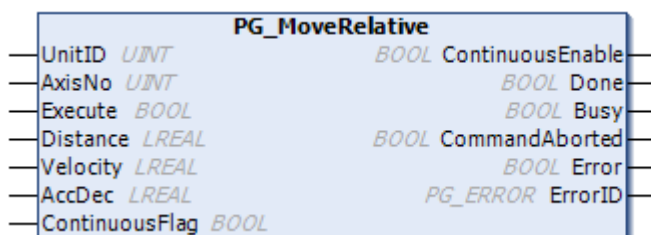


- When executing the function block, use the PG\_Power function block in advance to set to the servo ON state.
- If the function block is executed in the servo OFF state, the PG\_AXIS\_NOT\_READY\_FOR\_MOTION error is issued.

### 11.2.4 PG\_MoveRelative

This is a function block (FB) that causes the axis of the pulse output unit to travel to a position specified as a relative position.

#### ■ Icon



#### ■ Parameter

| Scope | Parameter name | Type  | Default | Description                           |
|-------|----------------|-------|---------|---------------------------------------|
| Input | UnitID         | UINT  | -       | Specifies the unit ID.                |
|       | AxisNo         | UINT  | -       | Specifies the axis No.                |
|       | Execute        | BOOL  | FALSE   | Starts execution at the rising edge.  |
|       | Distance       | LREAL | 0       | Specifies the relative distance (u).  |
|       | Velocity       | LREAL | 0       | Specifies the maximum velocity (u/s). |

## 11.2 Function Blocks for the Pulse Output Unit

| Scope  | Parameter name   | Type     | Default  | Description  |
|--------|------------------|----------|----------|--|
|        | AccDec           | LREAL    | 0        | Specifies the acceleration / deceleration ( $u/s^2$ ). |
|        | ContinuousFlag   | BOOL     | FALSE    | FALSE: E-point control<br>TRUE: P-point control        |
| Output | ContinuousEnable | BOOL     | FALSE    | TRUE: Distance, Velocity, and AccDec can be changed.   |
|        | Done             | BOOL     | FALSE    | TRUE: FB operation is completed.                       |
|        | Busy             | BOOL     | FALSE    | TRUE: FB operation is in progress.                     |
|        | CommandAborted   | BOOL     | FALSE    | TRUE: FB operation is interrupted.                     |
|        | Error            | BOOL     | FALSE    | TRUE: FB is abnormally completed.                      |
|        | ErrorID          | PG_ERROR | NO_ERROR | An error ID is output.                                 |

### ■ Operations when the function block is executed

- Relative value control is performed according to the distance specified in the Distance input. (Coordinates: -2147483648 to 2147483647u)
- To use the E-point control, specify as follows.
  - Specifies ContinuousFlag = FALSE and execute.
- To use the P-point control, specify as follows.
  - Specifies ContinuousFlag = TRUE and execute.
  - When the ContinuousEnable output becomes TRUE after PG\_MoveRelative is started, the Position, Velocity, and AccDec inputs can be rewritten. Change the inputs using the rising edge when the Execute input becomes TRUE as a trigger.
  - If rewriting is performed when the ContinuousEnable output is FALSE, the PG\_FB\_IN\_BUSY error is issued.
  - If rewriting is not executed, the execution of PG\_MoveRelative is completed (Done=TRUE) when the pulse output is completed.



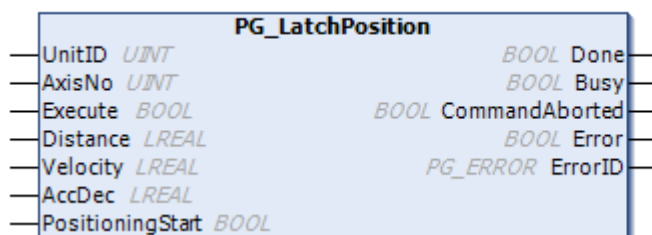
- When executing the function block, use the PG\_Power function block in advance to set to the servo ON state. If the function block is executed in the servo OFF state, the PG\_AXIS\_NOT\_READY\_FOR\_MOTION error is issued.

### 11.2.5 PG\_LatchPosition

This is a function block (FB) that causes the axis of the pulse output unit to travel to a position specified by an external signal input as a relative position.



### ■ Icon



### ■ Parameter

| Scope  | Parameter name   | Type     | Default  | Description  |
|--------|------------------|----------|----------|--|
| Input  | UnitID           | UINT     | -        | Specifies the unit ID.   |
|        | AxisNo           | UINT     | -        | Specifies the axis No.   |
|        | Execute          | BOOL     | FALSE    | Starts execution at the rising edge.                           |
|        | Distance         | LREAL    | 0        | Specifies the relative distance (u).                           |
|        | Velocity         | LREAL    | 0        | Specifies the maximum velocity (u/s).                          |
|        | AccDec           | LREAL    | 0        | Specifies the acceleration / deceleration (u/s <sup>2</sup> ). |
|        | PositioningStart | BOOL     | FALSE    | Positioning start input (for debugging)                        |
| Output | ContinuousEnable | BOOL     | FALSE    | TRUE: Start input  |
|        | Done             | BOOL     | FALSE    | TRUE: FB operation is completed.                               |
|        | Busy             | BOOL     | FALSE    | TRUE: FB operation is in progress.                             |
|        | CommandAborted   | BOOL     | FALSE    | TRUE: FB operation is interrupted.                             |
|        | Error            | BOOL     | FALSE    | TRUE: FB is abnormally completed.                              |
|        | ErrorID          | PG_ERROR | NO_ERROR | An error ID is output.   |

### ■ Operations when the function block is executed

- Latch operation is started by using the timing when the Execute input becomes TRUE as a trigger. Makes a relative movement starting from the input of the external signal for a distance specified in the Distance input.
- PositioningStart works as an input for the position control start input of the external signal. It can be used when performing a debug.



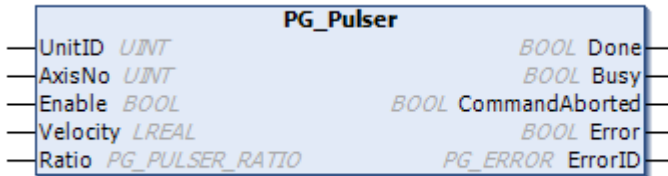
- When executing the function block, use the PG\_Power function block in advance to set to the servo ON state. If the function block is executed in the servo OFF state, the PG\_AXIS\_NOT\_READY\_FOR\_MOTION error is issued.
- The input parameter cannot be changed while this function block is executed (Busy = TRUE). Using the timing when the Execute input becomes TRUE as a trigger, the PG\_FB\_IN\_BUSY error is issued.

## 11.2 Function Blocks for the Pulse Output Unit

### 11.2.6 PG\_Pulser

This is a function block (FB) that enables constant speed operation for the axes of the pulse output unit using external pulse input.

#### ■ Icon



#### ■ Parameter

| Scope  | Parameter name | Type            | Default  | Description  |
|--------|----------------|-----------------|----------|--|
| Input  | UnitID         | UINT            | -        | Specifies the unit ID.   |
|        | AxisNo         | UINT            | -        | Specifies the axis No.   |
|        | Enable         | BOOL            | FALSE    | TRUE: Enables pulser operation.<br>FALSE: Disables pulser operation.           |
|        | Velocity       | LREAL           | 0        | Specifies the maximum velocity (u/s).  |
|        | Ratio          | PG_PULSER_RATIO | RATIO_x1 | Specifies the multiplication ratio between the pulser input and pulser output. |
| Output | Done           | BOOL            | FALSE    | TRUE: FB operation is completed.   |
|        | Busy           | BOOL            | FALSE    | TRUE: FB operation is in progress.   |
|        | CommandAborted | BOOL            | FALSE    | TRUE: FB operation is interrupted.   |
|        | Error          | BOOL            | FALSE    | TRUE: FB is abnormally completed.  |
|        | ErrorID        | PG_ERROR        | NO_ERROR | An error ID is output.   |

#### ■ Operations when the function block is executed



- When executing the function block, use the PG\_Power function block in advance to set to the servo ON state. If the function block is executed in the servo OFF state, the PG\_AXIS\_NOT\_READY\_FOR\_MOTION error is issued.

#### ■ PG\_PULSER\_RATIO (Enumeration type)

For specifying the multiplication ratio between the pulser input and pulser output using PG\_PULSER\_RATIO, refer to the following table.

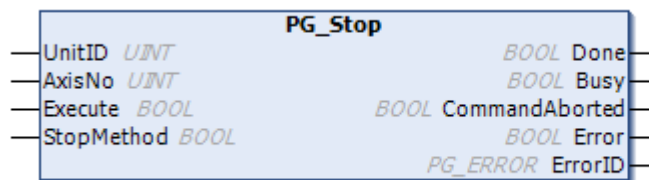
| Definition | Description |
|------------|-------------|
| RATIO_x1   | x1          |
| RATIO_x2   | x2          |

| Definition  | Description |
|-------------|-------------|
| RATIO_x5    | x5          |
| RATIO_x10   | x10         |
| RATIO_x50   | x50         |
| RATIO_x100  | x100        |
| RATIO_x500  | x500        |
| RATIO_x1000 | x1000       |

### 11.2.7 PG\_Stop

This is a function block (FB) that causes the axis of the pulse output unit to make a forced stop or deceleration stop.

#### ■ Icon



#### ■ Parameter

| Scope  | Parameter name | Type     | Default  | Description                                   |
|--------|----------------|----------|----------|---|
| Input  | UnitID         | UINT     | -        | Specifies the unit ID.                        |
|        | AxisNo         | UINT     | -        | Specifies the axis No.                        |
|        | Execute        | BOOL     | FALSE    | Starts execution at the rising edge.          |
|        | StopMethod     | BOOL     | FALSE    | TRUE: Forced stop<br>FALSE: Deceleration stop |
| Output | Done           | BOOL     | FALSE    | TRUE: FB operation is completed.              |
|        | Busy           | BOOL     | FALSE    | TRUE: FB operation is in progress.            |
|        | CommandAborted | BOOL     | FALSE    | TRUE: FB operation is interrupted.            |
|        | Error          | BOOL     | FALSE    | TRUE: FB is abnormally completed.             |
|        | ErrorID        | PG_ERROR | NO_ERROR | An error ID is output.                        |

#### ■ Operations when the function block is executed

- If PG\_Stop is executed while the following function block is being executed, execution of the function block that is being executed is interrupted (CommandAborted = TRUE).
  - PG\_Jog
  - PG\_MoveRelative

## 11.2 Function Blocks for the Pulse Output Unit

- PG\_MoveAbsolute
- PG\_LatchPosition
- PG\_Pulser
- PG\_Home

When executing one of these function blocks again, set the Execute input of PG\_Stop to FALSE and then execute the function block.

- The stop method can be specified using the StopMethod input.  
TRUE: Forced stop  
FALSE: Deceleration stop
- When the deceleration stop is selected, deceleration takes place at the acceleration / deceleration specified in function block.

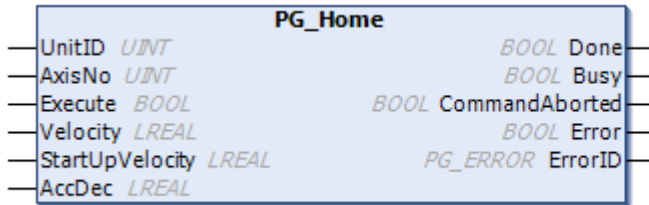


- When executing the function block, use the PG\_Power function block in advance to set to the servo ON state. If the function block is executed in the servo OFF state, the PG\_AXIS\_NOT\_READY\_FOR\_MOTION error is issued.
- The input cannot be changed while this function block is executed (Busy = TRUE). Using the timing when the Execute input becomes TRUE as a trigger, the PG\_FB\_IN\_BUSY error is issued.

### 11.2.8 PG\_Home

This is a function block (FB) that performs home return of the pulse output unit.

#### ■ Icon



#### ■ Parameter

| Scope  | Parameter name  | Type  | Default | Description  |
|--------|-----------------|-------|---------|--|
| Input  | UnitID          | UINT  | -       | Specifies the unit ID.   |
|        | AxisNo          | UINT  | -       | Specifies the axis No.   |
|        | Execute         | BOOL  | FALSE   | Starts execution at the rising edge.                           |
|        | Velocity        | LREAL | 0       | Specifies the maximum velocity (u/s).                          |
|        | StartUpVelocity | LREAL | 0       | Specifies the startup velocity (u/s).                          |
|        | AccDec          | LREAL | 0       | Specifies the acceleration / deceleration (u/s <sup>2</sup> ). |
| Output | Done            | BOOL  | FALSE   | TRUE: FB operation is completed.                               |

## 11.2 Function Blocks for the Pulse Output Unit

| Scope | Parameter name | Type     | Default  | Description                        |
|-------|----------------|----------|----------|------------------------------------|
|       | Busy           | BOOL     | FALSE    | TRUE: FB operation is in progress. |
|       | CommandAborted | BOOL     | FALSE    | TRUE: FB operation is interrupted. |
|       | Error          | BOOL     | FALSE    | TRUE: FB is abnormally completed.  |
|       | ErrorID        | PG_ERROR | NO_ERROR | An error ID is output.             |

### ■ Operations when the function block is executed

- When returning to the origin, the value of StartUpVelocity is used for the startup velocity, not the parameter setting.
- Use PG\_ReadStatus to read the external signal for the home input or the near home input.
- The StartUpVelocity should be 1 u/s or higher. The value set in the StartUpVelocity is the creep speed setting.
- The maximum velocity should be 1 u/s or more. If set to 0, an error will occur at runtime.

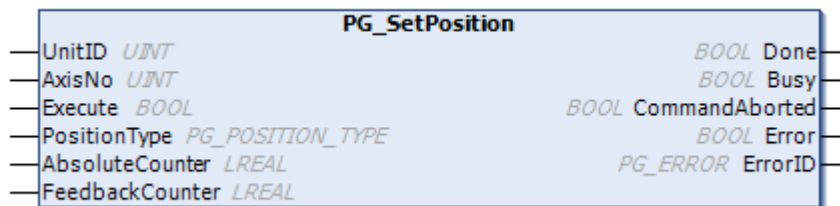


- When executing the function block, use the PG\_Power function block in advance to set to the servo ON state. If the function block is executed in the servo OFF state, the PG\_AXIS\_NOT\_READY\_FOR\_MOTION error is issued.
- The input cannot be changed while this function block is executed (Busy = TRUE). Using the timing when the Execute input becomes TRUE as a trigger, the PG\_FB\_IN\_BUSY error is issued.

### 11.2.9 PG\_SetPosition

This is a function block (FB) that sets the elapsed value and the feedback counter of the pulse output unit to desired values.

### ■ Icon



### ■ Parameter

| Scope | Parameter name | Type             | Default  | Description  |
|-------|----------------|------------------|----------|--|
| Input | UnitID         | UINT             | -        | Specifies the unit ID.                               |
|       | AxisNo         | UINT             | -        | Specifies the axis No.                               |
|       | Execute        | BOOL             | FALSE    | Starts execution at the rising edge.                 |
|       | PositionType   | PG_POSITION_TYPE | ABSOLUTE | Specify a setting target.<br>ABSOLUTE: Elapsed value |

## 11.2 Function Blocks for the Pulse Output Unit

| Scope  | Parameter name  | Type     | Default  | Description  |
|--------|-----------------|----------|----------|--|
|        |                 |          |          | FEEDBACK: Feedback counter<br>BOTH: Elapsed value and feedback counter |
|        | AbsoluteCounter | LREAL    | 0        | Set value (Elapsed value)  |
|        | FeedbackCounter | LREAL    | 0        | Set value (Feedback counter)   |
| Output | Done            | BOOL     | FALSE    | TRUE: FB operation is completed.                                       |
|        | Busy            | BOOL     | FALSE    | TRUE: FB operation is in progress.                                     |
|        | CommandAborted  | BOOL     | FALSE    | TRUE: FB operation is interrupted.                                     |
|        | Error           | BOOL     | FALSE    | TRUE: FB is abnormally completed.                                      |
|        | ErrorID         | PG_ERROR | NO_ERROR | An error ID is output.   |

### ■ Operations when the function block is executed

- When the PositionType input is set to BOTH, both the elapsed value and feedback counter can be set simultaneously.

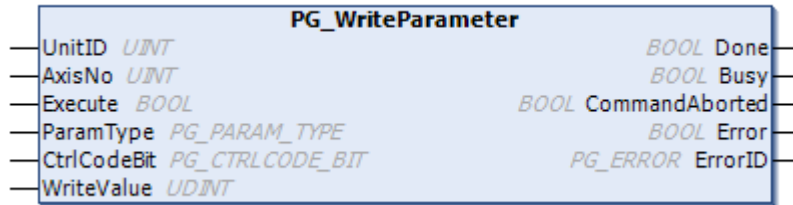


- Always execute the function while all axes are stopped. If the function block is executed while axes are moving, the PG\_AXIS\_IN\_DRIVEN error is issued.

### 11.2.10 PG\_WriteParameter

This is a function block (FB) that writes the parameters to the pulse output unit.

### ■ Icon



### ■ Parameter

| Scope | Parameter name | Type                | Default | Description   |
|-------|----------------|---------------------|---------|---|
| Input | UnitID         | UINT                | -       | Specifies the unit ID.  |
|       | AxisNo         | UINT                | -       | Specifies the axis No.  |
|       | Execute        | BOOL                | FALSE   | Starts execution at the rising edge.  |
|       | ParamType      | PG_PARAM_TY<br>PE   | 0       | Specify target parameters using PG_PARAM_TYPE and CtrlCodeBit in combination. |
|       | CtrlCodeBit    | PG_CTRLCODE<br>_BIT | 0       |   |

| Scope  | Parameter name | Type     | Default  | Description                        |
|--------|----------------|----------|----------|------------------------------------|
|        | WriteValue     | UDINT    | 0        | Write value                        |
| Output | Done           | BOOL     | FALSE    | TRUE: FB operation is completed.   |
|        | Busy           | BOOL     | FALSE    | TRUE: FB operation is in progress. |
|        | CommandAborted | BOOL     | FALSE    | TRUE: FB operation is interrupted. |
|        | Error          | BOOL     | FALSE    | TRUE: FB is abnormally completed.  |
|        | ErrorID        | PG_ERROR | NO_ERROR | An error ID is output.             |

### ■ Operations when the function block is executed

- If the GM1 is started, the values set in the "Pulse\_4Axes Parameters" of the GM Programmer are set.
- The parameters changed by PG\_WriteParameter are not saved in the pulse output unit.  
When saving the parameter data in the pulse output unit, use the held data. After starting the GM1, write parameters using PG\_WriteParameter.

### ■ How to Specify "Pulse\_4Axes Parameters"

Specify "Pulse\_4Axes Parameters" by combining the ParamType parameter and the CtrlCodeBit parameter as shown in the following table.

#### When ParamType = CONTROL\_CODE

- Specify a target parameter by combining the PARAM\_TYPE parameter and the CtrlCodeBit parameter.
- Specify a value to be written using the WriteValue parameter. Specify using PG\_CTRLCODE\_VALUE (enumeration type).

| Pulse_4Axes Parameter          | CtrlCodeBit                     | WriteValue                         | Description                 |
|--------------------------------|---------------------------------|------------------------------------|-----------------------------|
| DirectionOfRotationInput       | PULSE_INPUT_ROTATION_DIRECTION  | PULSEIN_DIRECTION_FORWARD          | Forward                     |
|                                |                                 | PULSEIN_DIRECTION_REVERSE          | Reverse                     |
| Count                          | PULSE_INPUT_COUNT               | PULSEIN_COUNT_ENABLE               | Enable                      |
|                                |                                 | PULSEIN_COUNT_DISABLE              | Disable                     |
| PulseInputMode                 | PULSE_INPUT_MODE                | PULSEIN_MODE_2PHASE                | 2-phase input               |
|                                |                                 | PULSEIN_MODE_DIRECTION_DISTINCTION | Direction distinction input |
|                                |                                 | PULSEIN_MODE_INDIVIDUAL            | Individual input            |
| PulseInputCount Multiplication | PULSE_INPUT_MULTIPLICATION      | PULSEIN_MULTIPLICATION_x1          | x1                          |
|                                |                                 | PULSEIN_MULTIPLICATION_x2          | x2                          |
|                                |                                 | PULSEIN_MULTIPLICATION_x4          | x4                          |
| DirectionOfRotationOutput      | PULSE_OUTPUT_ROTATION_DIRECTION | PULSEOUT_DIRECTION_FORWARD         | Forward                     |
|                                |                                 | PULSEOUT_DIRECTION_REVERSE         | Reverse                     |
| PulseOutputMode                | PULSE_OUTPUT_MODE               | PULSEOUT_MODE_PULSESIGN            | Pulse/Sign                  |

## 11.2 Function Blocks for the Pulse Output Unit

| Pulse_4Axes Parameter            | CtrlCodeBit                  | WriteValue                        | Description                        |
|----------------------------------|------------------------------|-----------------------------------|------------------------------------|
|                                  |                              | PULSEOUT_MODE_CWCCW               | CW/CCW                             |
| PulseOutputFrequencyDivisionMode | PULSE_OUTPUT_DIVIDED_MODE    | PULSEOUT_DIVIDED_BY1              | Divided by 1                       |
|                                  |                              | PULSEOUT_DIVIDED_BY2              | Divided by 2                       |
|                                  |                              | PULSEOUT_DIVIDED_BY4              | Divided by 4                       |
|                                  |                              | PULSEOUT_DIVIDED_BY8              | Divided by 8                       |
|                                  |                              | PULSEOUT_DIVIDED_BY16             | Divided by 16                      |
|                                  |                              | PULSEOUT_DIVIDED_BY32             | Divided by 32                      |
|                                  |                              | PULSEOUT_DIVIDED_BY64             | Divided by 64                      |
|                                  |                              | PULSEOUT_DIVIDED_BY128            | Divided by 128                     |
| DeviationCounterClearTime        | DEVIATION_COUNTER_CLEAR_TIME | DEVIATION_COUNTER_CLEAR_TIME_1ms  | 1ms                                |
|                                  |                              | DEVIATION_COUNTER_CLEAR_TIME_10ms | 10 ms                              |
| AccelerationDecelerationMethod   | PULSE_OUT_ACC_DEC            | PULSEOUT_ACC_DEC_LINEAR           | Linear Acceleration / Deceleration |
|                                  |                              | PULSEOUT_ACC_DEC_SSHAPED          | S Acceleration / Deceleration      |
| OriginReturnDirection            | HOMING_DIRECTION             | HOME_DIRECTION_NEGATIVE           | (-) Direction of the elapsed value |
|                                  |                              | HOME_DIRECTION_POSITIVE           | (+) Direction of the elapsed value |
| StartUpTime                      | STARTUP_TIME                 | STARTUP_TIME_20us                 | 0.02ms                             |
|                                  |                              | STARTUP_TIME_5us                  | 0.005ms                            |
|                                  |                              | STARTUP_TIME_1us                  | 0.001ms                            |
| OriginInputLogic                 | HOME_INPUT_LOGIC             | HOME_INPUT_NORMAL_CLOSE           | Normal Close                       |
|                                  |                              | HOME_INPUT_NORMAL_OPEN            | Normal Open                        |
| OriginNeighborhoodLogic          | NEARHOME_INPUT_LOGIC         | NEARHOME_INPUT_NORMAL_OPEN        | Normal Open                        |
|                                  |                              | NEARHOME_INPUT_NORMAL_CLOSE       | Normal Close                       |
| OriginSearch                     | HOME_SEARCH                  | HOME_SEARCH_DISABLE               | Disable                            |
|                                  |                              | HOME_SEARCH_ENABLE                | Enable                             |
| LimitInputLogic                  | LIMIT_INPUT_LOGIC            | LIMIT_INPUT_NORMAL_CLOSE          | Normal Close                       |
|                                  |                              | LIMIT_INPUT_NORMAL_OPEN           | Normal Open                        |
| SShapedPattern                   | S_ACC_DEC                    | S_ACC_DEC_SIN_CURVE               | Sin curve                          |
|                                  |                              | S_ACC_DEC_SECONDARY_CURVE         | Quadratic curve                    |
|                                  |                              | S_ACC_DEC_CYCLOID_CURVE           | Cycloid curve                      |
|                                  |                              | S_ACC_DEC_THIRD_CURVE             | Third curve                        |



### When other than ParamType = CONTROL\_CODE

- Specify a target parameter by using only the PARAM\_TYPE parameter.
- Specify a value to be written using the WriteValue parameter. Specify in the UDINT format.

| Pulse_4Axes Parameter          | ParamType                 | CtrlCodeBit | WriteValue   |
|--------------------------------|---------------------------|-------------|--|
| PulseInputAorB<br>SignalInCnst | PULSE_INPUT_SIGNAL_INCNST | -           | 0: Not InCnst (No input time constant)<br>1: 0.1 us<br>2: 0.5 us<br>3: 1.0 us<br>4: 2.0 us<br>5: 10.0 us |
| OriginInputInCnst              | HOME_INPUT_INCNST         | -           | 0: Not InCnst (No input time constant)<br>1: 10 us<br>2: 100 us  |

### 11.2.11 PG\_ReadParameter

This is a function block (FB) that reads the parameters of the pulse output unit.

#### ■ Icon



#### ■ Parameter

| Scope  | Parameter name | Type            | Default | Description   |
|--------|----------------|-----------------|---------|---|
| Input  | UnitID         | UINT            | -       | Specifies the unit ID.  |
|        | AxisNo         | UINT            | -       | Specifies the axis No.  |
|        | Execute        | BOOL            | FALSE   | Starts execution at the rising edge.  |
|        | ParamType      | PG_PARAM_TYPE   | 0       | Specify target parameters using PG_PARAM_TYPE and CtrlCodeBit in combination. |
|        | CtrlCodeBit    | PG_CTRLCODE_BIT | 0       |   |
| Output | ReadValue      | UDINT           | 0       | Read value  |
|        | Done           | BOOL            | FALSE   | TRUE: FB operation is completed.  |
|        | Busy           | BOOL            | FALSE   | TRUE: FB operation is in progress.  |

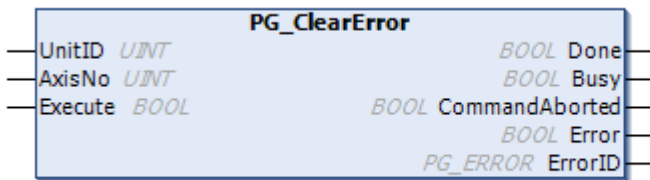
## 11.2 Function Blocks for the Pulse Output Unit

| Scope | Parameter name | Type     | Default  | Description                       |
|-------|----------------|----------|----------|-----------------------------------|
|       | Error          | BOOL     | FALSE    | TRUE: FB is abnormally completed. |
|       | ErrorID        | PG_ERROR | NO_ERROR | An error ID is output.            |

### 11.2.12 PG\_ClearError

This is a function block (FB) that clears the limit error or the set value error of the pulse output unit.

#### ■ Icon



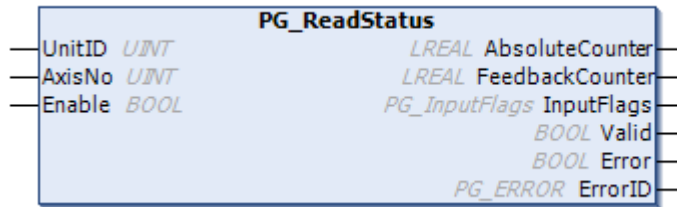
#### ■ Parameter

| Scope  | Parameter name | Type     | Default | Description                          |
|--------|----------------|----------|---------|--------------------------------------|
| Input  | UnitID         | UINT     | -       | Specifies the unit ID.               |
|        | AxisNo         | UINT     | -       | Specifies the axis No.               |
|        | Execute        | BOOL     | FALSE   | Starts execution at the rising edge. |
| Output | Done           | BOOL     | FALSE   | TRUE: FB operation is completed.     |
|        | Busy           | BOOL     | FALSE   | TRUE: FB operation is in progress.   |
|        | CommandAborted | BOOL     | FALSE   | TRUE: FB operation is interrupted.   |
|        | Error          | BOOL     | FALSE   | TRUE: FB is abnormally completed.    |
|        | ErrorID        | PG_ERROR | 0       | An error ID is output.               |

### 11.2.13 PG\_ReadStatus

This is a function block (FB) that reads the status of the pulse output unit.

### ■ Icon



### ■ Parameter

| Scope  | Parameter name  | Type          | Default  | Description                           |
|--------|-----------------|---------------|----------|---------------------------------------|
| Input  | UnitID          | UINT          | -        | Specifies the unit ID.                |
|        | AxisNo          | UINT          | -        | Specifies the axis No.                |
|        | Enable          | BOOL          | FALSE    | TRUE: Execution of the FB is enabled. |
| Output | AbsoluteCounter | LREAL         | 0        | Read value (Elapsed value)            |
|        | FeedbackCounter | LREAL         | 0        | Read value (Feedback counter)         |
|        | InputFlags      | PG_InputFlags | 0        | Read value content (Input flag)       |
|        | Valid           | BOOL          | FALSE    | TRUE:                                 |
|        | Error           | BOOL          | FALSE    | TRUE: FB is abnormally completed.     |
|        | ErrorID         | PG_ERROR      | NO_ERROR | An error ID is output.                |

### ■ PG\_InputFlags

For the contents of PG\_InputFlags, refer to the following table.

| Parameter                  | Name                | Description  |
|----------------------------|---------------------|--|
| PulseOutputBusy            | Pulse output busy   | TRUE when the pulse output is being output                                       |
| PulseOutputDone            | Pulse output done   | TRUE when the pulse output is completed  |
| AccelerationZone           | Acceleration zone   | TRUE when in the acceleration zone   |
| ConstantSpeedZone          | Constant speed zone | TRUE when in the constant speed zone   |
| DecelerationZone           | Deceleration zone   | TRUE when in the deceleration zone   |
| RotationDirection          | Rotation direction  | Monitors the rotation direction.<br>TRUE when the elapsed value is incrementing  |
| HomeInput                  | Home input          | Monitors the home Input signal.<br>TRUE when the home input is enabled           |
| NearHomeInput              | Near home input     | Monitors the near home input signal.<br>TRUE when the near home input is enabled |
| HomingDone                 | Home return done    | TRUE when the home return is completed   |
| OutputStopError            | Output stop error   | TRUE when an error occurs in the pulse output unit and when output is stopped    |
| SetValueChangeConfirmation | Set value change    | Used to confirm rewriting of the set value when P-point control is performed.    |

## 11.2 Function Blocks for the Pulse Output Unit

---

| Parameter              | Name                   | Description   |
|------------------------|------------------------|---|
| OverPositiveLimitInput | Over limit input (+)   | Monitor contact of the over limit input (+)<br>TRUE when over limit input (+) is enabled  |
| OverNegativeLimitInput | Over limit input (-)   | Monitor contact of the over limit input (-)<br>TRUE when over limit input (-) is enabled  |
| TimingInputMonitor     | Timing input monitor   | Monitor contact of the positioning control start input (timing input)<br>TRUE when the positioning control start input is enabled |
| SetValueError          | Set value error        | TRUE when a set value error occurs  |
| LimitError             | Limit error            | TRUE when a limit input is input during operation or when started   |
| ServoOnOutputState     | Servo ON output status | TRUE when servo ON  |

## 11.3 Error Codes

This section describes errors that are output in function blocks for pulse output unit and their contents. These errors are defined in PG\_ERROR.

### 11.3.1 Error Check Method

With a function block that has the Error and ErrorID output parameters, it is possible to check whether an error has occurred.

When the Error output becomes TRUE, its error content is output to the ErrorID.

#### ■ Error occurrence example

In the following example, PG\_NOT\_SUPPORTED has occurred in the PG\_Power function block.

Error = TRUE (An error has occurred.)

ErrorID = PG\_NOT\_SUPPORTED (The function block not supported in the simulation mode was executed.)

Declaration section

| Device.Application.MC_PRG |          |                  |
|---------------------------|----------|------------------|
| Expression                | Type     | Value            |
| PG_Power_0                | PG_Power |                  |
| bRegulatorRealState       | BOOL     | FALSE            |
| Busy                      | BOOL     | FALSE            |
| Error                     | BOOL     | TRUE             |
| ErrorID                   | PG_ERROR | PG_NOT_SUPPORTED |

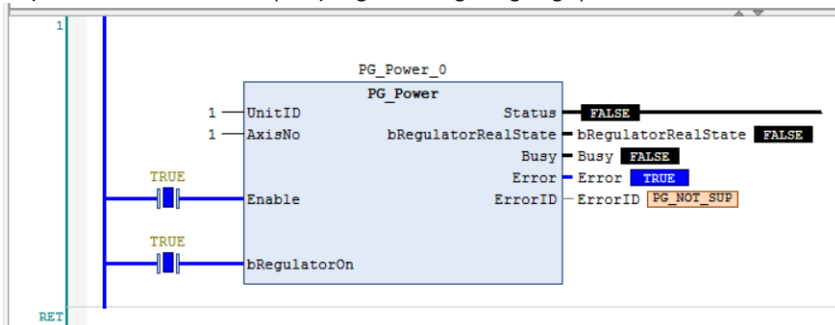
Implementation section (ST programming language)

```

1 PG_Power_0(UnitID:=1,
2   AxisNo:=1,
3   Enable:=TRUE,
4   bRegulatorOn:=TRUE,
5   Status:=FALSE,
6   bRegulatorRealState:=FALSE,
7   Busy:=FALSE,
8   Error:=TRUE,
9   ErrorID:=PG_NOT_SUP);RETURN

```

Implementation section (LD programming language)



## 11.3 Error Codes

### 11.3.2 PG\_ERROR

For the content of PG\_ERROR output in each function block, refer to the following table.

| Error name                   | Description   |
|------------------------------|---|
| PG_NO_ERROR                  | Normal (no error)   |
| PG_INVALID_UNIT              | The specified unit ID or axis No. is invalid.   |
| PG_SYSTEM_ERROR              | This is an internal error in the GM1 Controller.  |
| PG_NOT_SUPPORTED             | The function block not supported in the simulation mode was executed.   |
| PG_AXIS_NOT_READY_FOR_MOTION | The axis is in the servo OFF state.   |
| PG_AXIS_UNIT_ERROR           | The output of the pulse output unit has made an error stop.   |
| PG_AXIS_SET_VALUE_ERROR      | A set value error has occurred in the pulse output unit.  |
| PG_AXIS_LIMIT_ERROR          | A limit error has occurred in the pulse output unit.  |
| PG_AXIS_IN_STOP              | The function block could not be executed because stop processing was being executed by PG_Stop.                             |
| PG_AXIS_IN_RESET             | The function block could not be executed because axis information was being changed by PG_ClearError and PG_SetPosition.    |
| PG_AXIS_IN_DRIVEN            | The function block could not be executed because the axis was moving.   |
| PG_CHANGED_DURING_OPERATION  | The unit ID or axis No. was changed during operation.   |
| PG_INVALID_TARGET_VALUE      | Abnormal Information (position, speed, acceleration, or deceleration) was Input.  |
| PG_JOG_INVALID_REQUEST       | Both JogForward and JogBackward of JOG were simultaneously set to TRUE.   |
| PG_SET_VALUE_CHANGE_FAILED   | Failed to change command information.   |
| PG_FB_IN_BUSY                | Using the timing when Execute is set to TRUE as the trigger is invalid for the function block being executed (Busy = TRUE). |
| PG_ERROR_CLEAR_FAILED        | Failed to clear the error in the pulse output unit.   |
| PG_PARAMETER_WRITE_FAILED    | Failed to write the parameter.  |

# 12 Reference Information

---

|  |       |
|--|-------|
| 12.1 Motion Errors (SMC_ERROR Type) .....                        | 12-2  |
| 12.1.1 Error Check Method .....                                  | 12-2  |
| 12.1.2 SMC_ERROR .....   | 12-3  |
| 12.2 RTEX communication error .....                              | 12-11 |
| 12.2.1 RTEX Error ID .....                                       | 12-11 |
| 12.2.2 Alarm Codes .....   | 12-14 |
| 12.2.3 Warning Codes .....                                       | 12-18 |
| 12.3 List of AMP Parameters .....                                | 12-21 |
| 12.3.1 Class 0: Basic Setting .....                              | 12-21 |
| 12.3.2 Class 1: Gain Adjustment .....                            | 12-21 |
| 12.3.3 Class 2: Vibration Suppression Function .....             | 12-22 |
| 12.3.4 Class 3: Speed, Torque Control, Full-closed Control ..... | 12-23 |
| 12.3.5 Class 4: I/O Monitor Setting .....                        | 12-24 |
| 12.3.6 Class 5: Enhancing Setting .....                          | 12-25 |
| 12.3.7 Class 6: Special Setting 1 .....                          | 12-27 |
| 12.3.8 Class 7: Special Setting 2 .....                          | 12-29 |
| 12.3.9 Class 8: Special Setting 3 .....                          | 12-31 |
| 12.4 Monitor Commands .....                                      | 12-32 |

## 12.1 Motion Errors (SMC\_ERROR Type)

### 12.1 Motion Errors (SMC\_ERROR Type)

This section describes errors that are output in motion control instructions and their contents. Motion control errors are defined in SMC\_ERROR.

#### 12.1.1 Error Check Method

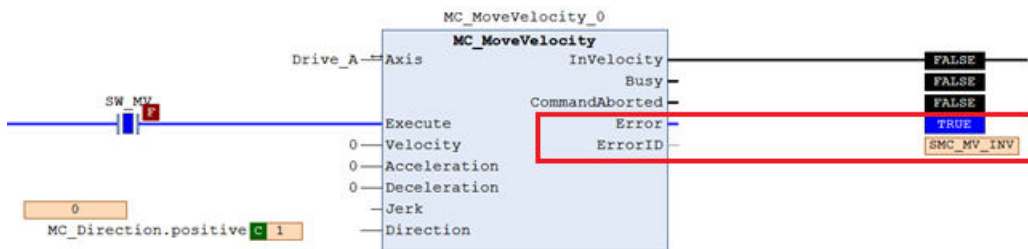
This section describes errors that are output in motion control instructions and their contents. Motion control errors are defined in SMC\_ERROR.

##### ■ Error check method

With a function block that has an output Error and output ErrorID, it is possible to check that an error has occurred.

The following shows an example of an error that has occurred when the MC\_MoveVelocity function block is executed.

“TRUE” is output to the output Error and an error is output to the output ErrorID.



The error name can be checked by double-clicking the output ErrorID.

An error name defined in the enumeration type SMC\_ERROR is displayed in the "Current value" field in the "Presetting values" dialog box.

Double-clicking `SMC_MV_INV` in the above execution example displays the following dialog box where error name "SMC\_MV\_INVALID\_ACCDEC\_VALUES" can be checked.



When an error occurs, the value of the error that has occurred (SMC\_ERROR) is also recorded in hexadecimal number on the "Log" screen of the device editor.

The following example shows a record when an error ("SMC\_MV\_INVALID\_ACCDEC\_VALUES") with an error value of 12D (301 in decimal) has occurred.

| Severity    | Time Stamp              | Description   | Component  |
|-------------|-------------------------|---|------------|
| Information | 10.11.2020 12:55:10.489 | ErrorInstance = 12D Application.MC_PRG.MC_MoveVelocity_0                        | SoftMotion |
| Error       | 10.11.2020 12:55:10.489 | FBEError Drive= 12D Invalid velocity, acceleration, deceleration or jerk values | SoftMotion |
| Information | 10.11.2020 12:55:04.375 | Application [Application] denied to start event                                 | CmpApp     |

### 12.1.2 SMC\_ERROR

| Error name                         | Value | Description   |
|------------------------------------|-------|---|
| SMC_NO_ERROR                       | 0     | No error  |
| SMC_DI_GENERAL_COMMUNICATION_ERROR | 1     | Communication error<br>Communication disconnection or another communication problem occurred. |
| SMC_DI_AXIS_ERROR                  | 2     | Axis error<br>Amplifier alarm or another axis problem occurred.                               |
| SMC_DI_FIELDBUS_LOST_SYNCRONICITY  | 3     | The fieldbus lost synchronicity.  |
| SMC_DI_SWLIMITS_EXCEEDED           | 10    | The software limit has been exceeded.   |
| SMC_DI_HWLIMITS_EXCEEDED           | 11    | The hardware end switch is active.  |

## 12.1 Motion Errors (SMC\_ERROR Type)

| Error name                             | Value | Description   |
|--|-------|---|
| SMC_DI_LINEAR_AXIS_OUTOFRANGE          | 12    | An overflow occurred in the linear axis.  |
| SMC_DI_HALT_OR_QUICKSTOP_NOT_SUPPORTED | 13    | The drive state is set to Halt or the Quickstop is unsupported.   |
| SMC_DI_VOLTAGE_DISABLED                | 14    | No power is supplied to the drive.  |
| SMC_DI_IRREGULAR_ACTPOSITION           | 15    | This error is not used.   |
| SMC_DI_POSITIONLAGERROR                | 16    | Position lag error<br>The difference between the commanded position and actual position has exceeded the specified limit when position lag monitoring is active.                              |
| SMC_DI_HOMING_ERROR                    | 17    | Home return error   |
| SMC_REGULATOR_OR_START_NOT_SET         | 20    | Either the controller is disabled or the brake has been applied.<br>Servo OFF or another similar problem occurred during axis movement.   |
| SMC_WRONG_CONTROLLER_MODE              | 21    | The executed function block is set to unsupported controller mode (SMC_SetControllerMode).  |
| SMC_INVALID_ACTION_FOR_LOGICAL         | 25    | Invalid operation was performed on the logical axis.  |
| SMC_FB_WASNT_CALLED_DURING_MOTION      | 30    | The function block was not called on the POU while the motion instruction was being executed ("Busy").<br>The operation was stopped while the motion instruction was being executed ("Busy"). |
| SMC_AXIS_IS_NO_AXIS_REF                | 31    | The type of the AXIS_REF type variable is different.  |
| SMC_AXIS_REF_CHANGED_DURING_OPERATION  | 32    | The AXIS_REF variable was changed during operation.   |
| SMC_FB_ACTIVE_AXIS_DISABLED            | 33    | The axis became disabled (MC_Power.bRegulatorOn) during movement.   |
| SMC_AXIS_NOT_READY_FOR_MOTION          | 34    | The axis cannot execute a motion instruction (an attempt was made to execute a motion instruction with MC_Stop enabled, for example).   |
| SMC_AXIS_ERROR_DURING_MOTION           | 35    | An error (such as amplifier alarm) occurred during motion operation.  |
| SMC_VD_MAX_VELOCITY_EXCEEDED           | 40    | The maximum velocity (fMaxVelocity) was exceeded.   |
| SMC_VD_MAX_ACCELERATION_EXCEEDED       | 41    | The maximum acceleration (fMaxAcceleration) was exceeded.   |
| SMC_VD_MAX_DECELERATION_EXCEEDED       | 42    | The maximum deceleration (fMaxDeceleration) was exceeded.   |
| SMC_3SH_INVALID_VELACC_VALUES          | 50    | Either an invalid velocity or acceleration was specified.   |
| SMC_3SH_MODE_NEEDS_HWLIMIT             | 51    | For safety reasons, a request is made to invoke the mode in which the end switch is used.   |
| SMC_FRC_NO_FREE_HANDLE                 | 60    | There is no file open handle.   |

## 12.1 Motion Errors (SMC\_ERROR Type)

| Error name                             | Value | Description   |
|--|-------|---|
| SMC_MAC_INITIALIZATION_FAILED          | 65    | SMC_MultiAcyclicCommunicator initialization failed.   |
| SMC_MAC_INVALID_TASK_HANDLE            | 66    | There is an invalid handle for the axis.  |
| SMC_MAC_TOO_MANY_TASKS                 | 67    | There are too many tasks that use an axis generating SDO.   |
| SMC_MAC_ATOMIC_ADD_FAILED              | 68    | An attempt to add Atomic failed.  |
| SMC_SDO_INVALID_DATALENGTH             | 69    | An invalid data length (> 4) occurred due to SDO reading.   |
| SMC_SCM_NOT_SUPPORTED                  | 70    | An invalid controller mode was set for SMC_SetControllerMode.   |
| SMC_SCM_AXIS_IN_WRONG_STATE            | 71    | The controller mode cannot be changed in the current axis state (an attempt was made to execute SMC_SetControllerMode with MC_Stop enabled, for example). |
| SMC_SCM_INTERRUPTED                    | 72    | SMC_SetControllerMode was interrupted by MC_Stop or ErrorStop.  |
| SMC_ST_WRONG_CONTROLLER_MODE           | 75    | The motion instruction was executed in an incorrect controller mode.  |
| SMC_RAG_ERROR_DURING_STARTUP           | 80    | An error occurred when the axis group was started up.   |
| SMC_RAG_ERROR_AXIS_NOT_INITIALIZED     | 81    | The axis is not in the specified state.   |
| SMC_PP_WRONG_AXIS_TYPE                 | 85    | The function block does not support virtual axes or logical axes.   |
| SMC_PP_NUMBER_OF_ABSOLUTE_BITS_INVALID | 86    | The number of bits is invalid (between 8 and 32 bits).  |
| SMC_CGR_ZERO_VALUES                    | 90    | An invalid value was specified.   |
| SMC_CGR_DRIVE_POWERED                  | 91    | A gear parameter was changed while the drive was in operation.  |
| SMC_CGR_INVALID_POSPERIOD              | 92    | An invalid position (0 or less, or half or more than the bus bandwidth) was specified.  |
| SMC_CGR_POSPERIOD_NOT_INTEGRAL         | 93    | The modulo period is not an integer.  |
| SMC_P_FTASCCYCLE_EMPTY                 | 110   | There is no information in one cycle time. (fTaskCycle = 0)   |
| SMC_R_NO_ERROR_TO_RESET                | 120   | There is no error to be reset (MC_Reset was executed when there was no function block error, for example).  |
| SMC_R_DRIVE_DOESNT_ANSWER              | 121   | There is no response to an error reset.   |
| SMC_R_ERROR_NOT_RESETTABLE             | 122   | An error reset cannot be executed.  |
| SMC_R_DRIVE_DOESNT_ANSWER_IN_TIME      | 123   | Communication with the axis is not working.   |
| SMC_R_CANNOT_RESET_COMMUNICATION_ERROR | 124   | A reset cannot be executed due to a communication error.  |
| SMC_RP_PARAM_UNKNOWN                   | 130   | The parameter number is undefined.  |
| SMC_RP_REQUESTING_ERROR                | 131   | An error occurred in communication with the drive.  |
| SMC_RP_DRIVE_PARAMETER_NOT_MAPPED      | 132   | Parameters are not assigned to the drive.   |

## 12.1 Motion Errors (SMC\_ERROR Type)

| Error name                        | Value | Description   |
|-----------------------------------|-------|---|
| SMC_RP_PARAM_CONVERSION_ERROR     | 133   | Conversion of drive parameter values failed. Soft motion parameters are undefined.                          |
| SMC_WP_PARAM_INVALID              | 140   | The parameter number is undefined or write operations are inhibited.  |
| SMC_WP_SENDING_ERROR              | 141   | Refer to the error number for WriteDriveParameter.  |
| SMC_WP_DRIVE_PARAMETER_NOT_MAPPED | 142   | Parameters are undefined for the drive.   |
| SMC_WP_PARAM_CONVERSION_ERROR     | 143   | Conversion of drive parameter values failed. Soft motion parameters are undefined.                          |
| SMC_H_AXIS_WASNT_STANDSTILL       | 170   | The axis is not in a standstill state.  |
| SMC_H_AXIS_DIDNT_START_HOMING     | 171   | An error occurred when home return was started.   |
| SMC_H_AXIS_DIDNT_ANSWER           | 172   | An error occurred when home return was started.   |
| SMC_H_ERROR_WHEN_STOPPING         | 173   | An error occurred after the axis stopped in home return mode. It is possible that deceleration was not set. |
| SMC_H_AXIS_IN_ERRORSTOP           | 174   | The drive is in the ErrorStop state. Home return cannot be executed.  |
| SMC_MS_UNKNOWN_STOPPING_ERROR     | 180   | Undefined error   |
| SMC_MS_INVALID_ACCDEC_VALUES      | 181   | Either an invalid velocity or acceleration was specified.   |
| SMC_MS_DIRECTION_NOT_APPLICABLE   | 182   | "shortest" cannot be applied to the direction.  |
| SMC_MS_AXIS_IN_ERRORSTOP          | 183   | Because the drive is in the ErrorStop state, stop operation cannot be executed with MC_Stop.                |
| SMC_BLOCKING_MC_STOP_WASNT_CALLED | 184   | MC_Stop (Execute=TRUE) blocks the axis. MC_Stop (Execute=FALSE) must be executed.                           |
| SMC_UNKNOWN_TASK_INTERVAL         | 200   | The task interval of the bus task is undefined.   |
| SMC_MA_INVALID_VELACC_VALUES      | 201   | Either an invalid velocity or acceleration was specified.   |
| SMC_MA_INVALID_DIRECTION          | 202   | There is an error in the specified direction ("Direction").   |
| SMC_MR_INVALID_VELACC_VALUES      | 226   | Either an invalid velocity or acceleration was specified.   |
| SMC_MR_INVALID_DIRECTION          | 227   | There is an error in the specified direction ("Direction").   |
| SMC_MAD_INVALID_VELACC_VALUES     | 251   | Either an invalid velocity or acceleration was specified.   |
| SMC_MAD_INVALID_DIRECTION         | 252   | There is an error in the specified direction ("Direction").   |
| SMC_MSI_INVALID_VELACC_VALUES     | 276   | Either an invalid velocity or acceleration was specified.   |
| SMC_MSI_INVALID_DIRECTION         | 277   | There is an error in the specified direction ("Direction").   |
| SMC_MSI_INVALID_EXECUTION_ORDER   | 278   | The same instance of MC_MoveSuperImposed was called more than once in a single cycle.                       |

## 12.1 Motion Errors (SMC\_ERROR Type)

| Error name                        | Value | Description  |
|-----------------------------------|-------|--|
| SMC_LOGICAL_NO_REAL_AXIS          | 300   | Unused error   |
| SMC_MV_INVALID_ACCDEC_VALUES      | 301   | Either an invalid velocity or acceleration was specified.  |
| SMC_MV_DIRECTION_NOT_APPLICABLE   | 302   | "shortest" or "fastest" cannot be applied to the direction ("Direction").  |
| SMC_PP_ARRAYSIZE                  | 325   | There is an error in the specified array size.   |
| SMC_PP_STEP0MS                    | 326   | The step time is 0s.   |
| SMC_VP_ARRAYSIZE                  | 350   | There is an error in the specified array size.   |
| SMC_VP_STEP0MS                    | 351   | The step time is 0s.   |
| SMC_AP_ARRAYSIZE                  | 375   | There is an error in the specified array size.   |
| SMC_AP_STEP0MS                    | 376   | The step time is 0s.   |
| SMC_TP_TRIGGEROCCUPIED            | 400   | The trigger is already enabled.  |
| SMC_TP_COULDNT_SET_WINDOW         | 401   | The drive does not support the window function.  |
| SMC_TP_COMM_ERROR                 | 402   | Communication error  |
| SMC_AT_TRIGGERNOTOCCUPIED         | 410   | The trigger is already disabled.   |
| SMC_MCR_INVALID_VELACC_VALUES     | 426   | Either an invalid velocity or acceleration was specified.  |
| SMC_MCR_INVALID_DIRECTION         | 427   | An invalid direction was specified.  |
| SMC_MCA_INVALID_VELACC_VALUES     | 451   | Either an invalid velocity or acceleration was specified.  |
| SMC_MCA_INVALID_DIRECTION         | 452   | An invalid direction was specified.  |
| SMC_MCA_DIRECTION_NOT_APPLICABLE  | 453   | "fastest" cannot be applied to the direction ("Direction").  |
| SMC_SDL_INVALID_AXIS_STATE        | 475   | Function block "SMC_ChangeDynamicLimits" was executed in a state other than "Standstill" or "Power_off".             |
| SMC_SDL_INVALID_VELACC_VALUES     | 476   | An invalid velocity, acceleration, deceleration, or jerk was specified.  |
| SMC_CR_NO_TAPPETS_IN_CAM          | 600   | The cam is not equipped with a tappet.   |
| SMC_CR_TOO_MANY_TAPPETS           | 601   | The tappet group ID exceeds MAX_NUM_TAPPETS.   |
| SMC_CR_MORE_THAN_32_ACCESSES      | 602   | There are 32 or more accesses to one cam.  |
| SMC_CI_NO_CAM_SELECTED            | 625   | No cam is selected.<br>It is possible that the correct cam table is not set in the CamTableID parameter of MC_CamIn. |
| SMC_CI_MASTER_OUT_OF_SCALE        | 626   | The current commanded position on the master axis is outside the range of the cam table.                             |
| SMC_CI_RAMPIN_NEEDS_VELACC_VALUES | 627   | A velocity and acceleration must be specified when StartMode is set to ramp_in.                                      |
| SMC_CI_SCALING_INCORRECT          | 628   | The scaling variables (fEditor, TableMasterMin, and Max) are incorrect.  |
| SMC_CI_TOO_MANY_TAPPETS_PER_CYCLE | 629   | The number of tappet outputs is too many to be enabled in one cycle.   |

## 12.1 Motion Errors (SMC\_ERROR Type)

| Error name                                | Value | Description   |
|---|-------|---|
| SMC_CB_NOT_IMPLEMENTED                    | 640   | The function block for cam format is not implemented.                                     |
| SMC_GI_RATIO_DENOM                        | 675   | RatioDenominator (denominator of the gear ratio) is set to 0.                             |
| SMC_GI_INVALID_ACC                        | 676   | The value specified in "Acceleration" is invalid.   |
| SMC_GI_INVALID_DEC                        | 677   | The value specified in "Deceleration" is invalid.   |
| SMC_GI_MASTER_REGULATOR_CHANGED           | 678   | The Enable / Disable state of the master axis was changed without permission.             |
| SMC_GI_INVALID_JERK                       | 679   | The value specified in "Jerk" is invalid.   |
| SMC_PH_INVALID_VELACCDEC                  | 725   | The specified velocity, acceleration, or deceleration were invalid.                       |
| SMC_PH_ROTARYAXIS_PERIOD0                 | 726   | fPositionPeriod for the rotation axis is set to 0.  |
| SMC_NO_CAM_REF_TYPE                       | 750   | The cam type is not an MC_CAM_REF structure.  |
| SMC_CAM_TABLE_DOES_NOT_COVER_MASTER_SCALE | 751   | The master axis area (xStart and xEnd) of the cam table is outside the curve data range.  |
| SMC_CAM_TABLE_EMPTY_MASTER_RANGE          | 752   | The cam data table is empty.  |
| SMC_CAM_TABLE_INVALID_MASTER_MINMAX       | 753   | The maximum value and minimum value of the master axis in the cam data are invalid.       |
| SMC_CAM_TABLE_INVALID_SLAVE_MINMAX        | 754   | The maximum value and minimum value of the slave axis in the cam data are invalid.        |
| SMC_GIP_MASTER_DIRECTION_CHANGE           | 775   | The rotation direction of the master axis was changed while the slave axis was connected. |
| SMC_GIP_SLAVE_REVERSAL_CANNOT_BE_AVOIDED  | 776   | The AvoidReversal input is set, but reverse rotation of the slave axis cannot be avoided. |
| SMC_GIP_AVOID_REVERSAL_FOR_FINITE_AXES    | 777   | The AvoidReversal input cannot be set for the finite slave axis.                          |
| SMC_BC_BL_TOO_BIG                         | 800   | The fBacklash gear backlash is too large (larger than position period/2).                 |
| SMC_QPROF_DIVERGES                        | 825   | Internal error: Failed in calculating the secondary path                                  |
| SMC_QPROF_INVALID_PARAMETER               | 826   | Internal error: Failed in calculating the secondary path                                  |
| SMC_QPROF_NO_RESULT                       | 827   | Internal error: Failed in calculating the secondary path                                  |
| SMC_QPROF_INVALID_NEW_LBD                 | 828   | Internal error: Failed in calculating the secondary path                                  |
| SMC_QPROF_BAD_NEGOTIATION                 | 829   | Internal error: Failed in calculating the secondary path                                  |
| SMC_QPROF_INVALID_INTERVAL                | 830   | Internal error: Failed in calculating the secondary path                                  |
| SMC_QPROF_NOT_ENOUGH_PHASES               | 831   | Internal error: Failed in calculating the secondary path                                  |
| SMC_TG_INTERNAL_ERROR                     | 832   | Internal error: Failed in calculating the secondary path                                  |

## 12.1 Motion Errors (SMC\_ERROR Type)

| Error name                                     | Value | Description   |
|--|-------|---|
| SMC_SRT_NOT_STANDSTILL_OR_POWEROFF             | 850   | Execution is possible only in the standstill or power_off state.                        |
| SMC_SRT_INVALID_RAMPTYPE                       | 851   | The value specified in RampType is invalid.   |
| SMC_SMT_NOT_STANDSTILL_OR_POWEROFF             | 852   | Execution is possible only in the standstill or power_off state.                        |
| SMC_SMT_INVALID_MOVEMENTTYPE_OR_POSITIONPERIOD | 853   | The value specified in MovementType or PositionPeriod is invalid.                       |
| SMC_SMT_AXIS_NOT_VIRTUAL                       | 854   | The function block is valid only for the virtual axis.                                  |
| SMC_NO_LICENSE                                 | 1000  | License error   |
| SMC_INT_VEL_ZERO                               | 1001  | Because Velocity is set to 0, path processing cannot be performed.                      |
| SMC_INT_NO_STOP_AT_END                         | 1002  | The final velocity of the path is other than 0.   |
| SMC_INT_DATA_UNDERRUN                          | 1003  | GEOINFO-List was processed by DataIn, but the end of the list has not been reached.     |
| SMC_INT_VEL_NONZERO_AT_STOP                    | 1004  | The velocity at the time of stoppage is greater than 0.                                 |
| SMC_INT_TOO_MANY_RECURSIONS                    | 1005  | There are too many recursions of SMC_Interpolator.                                      |
| SMC_INT_NO_CHECKVELOCITIES                     | 1006  | SMC_CheckVelocities is not called by SMC_OUTQUEUE.                                      |
| SMC_INT_PATH_EXCEEDED                          | 1007  | Internal error or calculation error   |
| SMC_INT_VEL_ACC_DEC_ZERO                       | 1008  | The specified velocity and acceleration / deceleration are 0 or less.                   |
| SMC_INT_DWIPOTIME_ZERO                         | 1009  | The motion task was called when dwlpoTime = 0.  |
| SMC_INT_JERK_NONPOSITIVE                       | 1010  | A negative value was set for "Jerk".  |
| SMC_INT_QPROF_DIVERGES                         | 1011  | Internal error<br>The calculation algorithm is incorrect.                               |
| SMC_INT_INVALID_VELOCITY_MODE                  | 1012  | The specified velocity mode is invalid.   |
| SMC_INT_TOO_MANY_AXES_INTERPOLATED             | 1013  | The maximum allowable number of axes for interpolation has been exceeded.               |
| SMC_INT_DEGENERATE_SEGMENT                     | 1014  |   |
| SMC_INT2DIR_BUFFER_TOO_SMALL                   | 1050  |   |
| SMC_INT2DIR_PATH_FITS_NOT_IN_QUEUE             | 1051  |   |
| SMC_XINT_INVALID_DIRECTION                     | 1070  |   |
| SMC_XINT_NOINTERSECTION                        | 1071  |   |
| SMC_WAR_INT_OUTQUEUE_TOO_SMALL                 | 1080  |   |
| SMC_WAR_END_VELOCITIES_INCORRECT               | 1081  | The specified final velocity is incorrect.  |
| SMC_CV_ACC_DEC_VEL_NONPOSITIVE                 | 1100  | Negative values are specified for the velocity and acceleration/deceleration.           |
| SMC_CA_INVALID_ACCDEC_VALUES                   | 1120  | Negative values are specified for fGapVelocity, fGapAcceleration, and fGapDeceleration. |
| SMC_DEC_ACC_TOO_LITTLE                         | 1200  | The specified acceleration is unacceptable.   |

## 12.1 Motion Errors (SMC\_ERROR Type)

| Error name                          | Value | Description   |
|-------------------------------------|-------|---|
| SMC_DEC_RET_TOO_LITTLE              | 1201  | The specified deceleration is unacceptable.   |
| SMC_DEC_OUTQUEUE_RAN_EMPTY          | 1202  | Data underrun<br>The queue was read, but it was empty.                                    |
| SMC_DEC_JUMP_TO_UNKNOWN_LINE        | 1203  | Because the line number is unknown, the cursor cannot jump to the line.                   |
| SMC_DEC_INVALID_SYNTAX              | 1204  | The syntax is invalid.  |
| SMC_DEC_3DMODE_OBJECT_NOT_SUPPORTED | 1205  | The object is not supported in 3D mode.   |
| SMC_DEC_NEGATIVE_PERIOD             | 1206  | A negative value is specified for the period during which an additional axis is disabled. |
| SMC_DEC_DIMENSIONS_EXCLUSIVE_AU     | 1207  | Both axis A and axis U are not always interpolated.<br>PA and PU are mutually exclusive.  |
| SMC_DEC_DIMENSIONS_EXCLUSIVE_BV     | 1208  | Both axis B and axis V are not always interpolated.<br>PB and PV are mutually exclusive.  |
| SMC_DEC_DIMENSIONS_EXCLUSIVE_CW     | 1209  | Both axis C and axis W are not always interpolated.<br>PC and PW are mutually exclusive.  |
| SMC_IPR_TOO_SMALL_BUFFER            | 1259  | The buffer size specified for OutQueue is too small.                                      |
| SMC_GCV_BUFFER_TOO_SMALL            | 1300  |   |
| SMC_GCV_BUFFER_WRONG_TYPE           | 1301  |   |
| SMC_GCV_UNKNOWN_IPO_LINE            | 1302  |   |
| SMC_NO_CNC_REF_TYPE                 | 1500  |   |
| SMC_NO_OUTQUEUE_TYPE                | 1501  | The specified pointer is not SMC_OUTQUEUE.  |
| SMC_GEOINFO_BUFFER_MISALIGNED       | 1502  | The buffer segments aligned by four-byte boundaries are not used by pbyBuffer.            |
| SMC_3D_MODE_NOT_SUPPORTED           | 1600  | The FB functions only with 2D paths.  |



## 12.2 RTEX communication error

### 12.2.1 RTEX Error ID

#### ■ WARNING\_CODE (Union type)

| Member             | Type                    | Description   |
|--------------------|-------------------------|---|
| uiWarningCode      | UINT                    | Warning code  |
| tWarningCodeMember | ALARM_WARNING_C<br>ODES | Main code (warning number) and sub-code (0) of the warning code |

#### ■ List of RTEX Error IDs

| Category                           | Error_Code / Sub_Error_Code | Cause   |
|------------------------------------|-----------------------------|---|
| Command header related             | 0011h                       | <ul style="list-style-type: none"> <li>Mismatched node address (MAC-ID)</li> </ul>  |
|                                    | 0012h                       | <ul style="list-style-type: none"> <li>C/R bit is 1 despite of command.</li> <li>Sub_Chk is 0 in 32-byte mode.</li> </ul>   |
| Command code, control mode related | 0021h                       | <ul style="list-style-type: none"> <li>Cyclic command is not defined.</li> </ul>  |
|                                    | 0022h                       | <ul style="list-style-type: none"> <li>Non-cyclic command is not defined (when cyclic command is normal).</li> <li>Combination error of control mode and non-cyclic command.</li> <li>Subcommand is undefined in 32-byte mode.</li> </ul>   |
|                                    | 002Eh                       | <ul style="list-style-type: none"> <li>Combination of communication cycle, semi-closed/full-closed, 16 / 32 byte mode, and control mode is not correct.</li> <li>Control mode has been changed in less than 2 ms.</li> <li>Control mode has been changed during profile position latch positioning / profile home return (Type_Code = 12h, 13h, 31h, 32h, 33h, 34h, 36h) operation.</li> <li>Control mode has been changed during execution of non-cyclic command (Busy = 1).</li> <li>Run the home return command (□4h) Type_Code = □4h / 2□4h during the velocity control (CV) / torque control (CT).</li> <li>Control mode has been changed to the velocity control during the 2 degrees of freedom control (synchronous) mode.</li> <li>Control mode has been changed to the torque control during the 2 degrees of freedom control (standard / synchronous) mode.</li> <li>Control mode has been changed during the retracting operation.</li> </ul> |
| Argument related                   | 0031h                       | <ul style="list-style-type: none"> <li>Type_Code / Sub_Type_Code is not defined.</li> </ul>   |
|                                    | 0032h                       | <ul style="list-style-type: none"> <li>Non-cyclic data / sub-command data other than Type_Code / Sub_Type_Code is out of setup range.</li> </ul>  |
|                                    | 0033h                       | <ul style="list-style-type: none"> <li>Cyclic data (Command_Data1) is out of setup range.</li> </ul>  |
|                                    | 0034h                       | <ul style="list-style-type: none"> <li>Feed forward data (Command_Data3, Sub_Command_Data2 / 3) is out of setup range.</li> </ul>   |
| Not executable 1 (general)         | 0041h                       | <ul style="list-style-type: none"> <li>Write access is attempted to read only media.</li> </ul>   |
|                                    | 0042h                       | <ul style="list-style-type: none"> <li>Alarm clear command is executed while an alarm that cannot be cleared has occurred and no warning was issued.</li> </ul>   |

## 12.2 RTEX communication error

| Category                                     | Error_Code / Sub_Error_Code | Cause   |
|--|-----------------------------|---|
|  | 0043h                       | <ul style="list-style-type: none"> <li>External scale error clear command is executed when not in full-closed control mode or when no external scale error is detected.</li> </ul>  |
|  | 0045h                       | <ul style="list-style-type: none"> <li>In servo on state, reset command is executed in attribute C parameter validation mode.</li> </ul>  |
|  | 0046h                       | <ul style="list-style-type: none"> <li>After deceleration and stop according to the drive inhibit input (POT / NOT), direction command POT / NOT is applied.</li> <li>During deceleration according to the drive inhibit input (POT / NOT), a profile operation (except Type_Code = 31h, 32h, 33h, 34h, and 36h) is started.</li> </ul>   |
| Not executable 2<br>(related to home return) | 0051h                       | <ul style="list-style-type: none"> <li>Multi-turn clearing of the home return command was executed while the encoder was in the incremental mode.</li> <li>Multi-turn clearing of the home return command was executed even when the single-turn absolute function was effective.</li> </ul>  |
|  | 0052h                       | <ul style="list-style-type: none"> <li>During cyclic position control (CP) (* including full-closed control) in absolute mode, Type_Code = 1□h of the home return command (□4h) has been executed.</li> <li>During profile position control (PP) (* including full-closed control) in absolute mode, profile home return has been executed.</li> </ul>  |
|  | 0053h                       | <ul style="list-style-type: none"> <li>During cyclic position control (CP) (* including full-closed control) in absolute mode, actual position set / command position set (Type_Code = 21h, 22h) of the home return command (□4h) have been executed.</li> </ul>  |
|  | 0055h                       | <ul style="list-style-type: none"> <li>Multi-turn clearing of the home return command is executed while in the full-closed control mode.</li> </ul>   |
|  | 0056h                       | <ul style="list-style-type: none"> <li>Multi-turn clearing of the home return command is executed while in the servo-on condition.</li> </ul>   |
|  | 0057h                       | <ul style="list-style-type: none"> <li>Type_Code = 1□h of the home return command is executed while in the servo-off state.</li> </ul>  |
|  | 0058h                       | <ul style="list-style-type: none"> <li>While the external input is not assigned to the latch correction terminal, Type_Code is executed by using the external input as a trigger.</li> <li>Started the latch mode with a stop function operated by the amplifier output signal as the trigger signal when Pr7.111 "Trigger signal assignment setting for the latch mode with a stop function" was set to 0 "Disabled".</li> </ul>   |
|  | 0059h                       | <ul style="list-style-type: none"> <li>Executed the home return command (□4h) while the profile position latch positioning / profile home return (Type_Code = 12h, 13g, 31h, 32h, 33h, 34h, 36h) was operated.</li> <li>During profile positioning / profile continuous revolution (Type_Code = 10h, 11h, 20h), initialization mode (Type_Code = 1□h, 31h) of home return command (□4h) has been executed.</li> </ul>   |
|  | 005Ah                       | <ul style="list-style-type: none"> <li>Z phase is set to latch trigger signal despite absolute external scale.</li> </ul>   |
|  | 005Bh                       | <ul style="list-style-type: none"> <li>Received the following commands in the virtual full-closed control mode. <ul style="list-style-type: none"> <li>Home return command (□4h)</li> <li>Profile position latch absolute positioning (12h) of the profile command (17h)</li> <li>Profile position latch absolute positioning (13h) of the profile command (17h)</li> <li>Profile home return (31h to 34h, 36h) of the profile command (17h)</li> <li>Config command</li> </ul> </li> </ul> |

| Category   | Error_Code / Sub_Error_Code | Cause  |
|--|-----------------------------|--|
|  |                             | <ul style="list-style-type: none"> <li>Received a command to change to the virtual full-closed control mode under the following conditions. <ul style="list-style-type: none"> <li>While initialization mode of home return command (□4h) was operated, latch mode was operated, or latch mode with stop function was operated</li> <li>Changed to a command other than command code (□4h) after starting home return command (Type_Code: 51h to 53h)</li> <li>During a period from starting the latch to detecting the latch after starting home return command (Type_Code: 51h to 53h)</li> <li>While profile position latch absolute positioning (12h) of the profile command (17h) was operated</li> <li>While profile position latch absolute positioning (13h) of the profile command (17h) was operated</li> <li>While profile home return (31h to 34h, 36h) of the profile command (17h) was operated</li> <li>After starting profile command (12h, 13h, 31h to 34h, 36h), during the period from when a change was made to a command other than command code (17h) until the latch or home was detected</li> <li>While Config command was executed</li> </ul> </li> </ul> |
|  | 005Fh                       | <ul style="list-style-type: none"> <li>Latch mode with stop function (Type_Code = F1h) was used in a setting other than the cyclic position control (CP).</li> <li>Latch mode with stop function (Type_Code = F1h) was used in a setting other than the communication cycle of 0.5 ms/command update cycle of 1.0 ms</li> <li>Latch mode with stop function (Type_Code = F1h) was used in a setting other than the electronic gear ratio of less than 1.</li> </ul>  |
| Not executable 3<br>(Related to hardware factor) | 0061h                       | <ul style="list-style-type: none"> <li>EEPROM writing is not permitted because of under voltage of the control power.</li> </ul>   |
| Not executable 4<br>(in process)                 | 0101h                       | <ul style="list-style-type: none"> <li>Not permitted to be accepted because the previous command is in process.</li> </ul>   |
|  | 0102h                       | <ul style="list-style-type: none"> <li>Command is not permitted to be accepted because the servo driver is accessing to the encoder now.</li> </ul>  |
|  | 0103h                       | <ul style="list-style-type: none"> <li>Command is not permitted to be accepted because the servo driver is accessing to the external scale now.</li> </ul>   |
|  | 0104h                       | <ul style="list-style-type: none"> <li>Type_Code has been changed while operating under profile position control (PP).</li> </ul>  |
|  | 0105h                       | <ul style="list-style-type: none"> <li>During execution of the PANATERM command (test run operation, FFT, Z phase search, pin assignment setting, or fit gain), received the RTEX command (reset command, home return command, or parameter command).</li> </ul>   |
| Not executable 5<br>(access inhibited)           | 0201h                       | <ul style="list-style-type: none"> <li>Command is not permitted to be accepted because parameter writing or writing to EEPROM is inhibited now.</li> <li>Write parameter command or write EEPROM command is issued while bit 0 of Pr7.23 RTEX function expansion setup 2 is set at 1.</li> </ul>   |

## 12.2 RTEX communication error

### 12.2.2 Alarm Codes

#### ■ ALARM\_CODE (Union type)

| Member           | Type                    | Description                              |
|------------------|-------------------------|--|
| uiAlarmCode      | UINT                    | Alarm code                               |
| tAlarmCodeMember | ALARM_WARNING_C<br>ODES | Main code and sub-code of the alarm code |

#### ■ List of alarm codes

| Error No. |     | Alarm name   | Attribute |                       |                                    |
|-----------|-----|--|-----------|-----------------------|------------------------------------|
| Main      | Sub |  | History   | Can be cleared        | Emergency stop <sup>(Note 6)</sup> |
| 11        | 0   | Control power supply undervoltage protection                                     |           | ○                     |                                    |
| 12        | 0   | Over-voltage protection  | ○         | ○                     |                                    |
| 13        | 0   | Main power supply undervoltage protection (Insufficient voltage between P and N) |           | ○                     | ○                                  |
|           | 1   | Main power supply undervoltage protection (AC interception detection)            |           | ○                     | ○                                  |
| 14        | 0   | Over-current protection  | ○         |                       |                                    |
|           | 1   | IPM error protection   | ○         |                       |                                    |
| 15        | 0   | Overheat protection  | ○         |                       | ○                                  |
|           | 1   | Encoder overheat error protection  | ○         |                       | ○                                  |
| 16        | 0   | Overload protection  | ○         | ○ <sup>(Note 1)</sup> |                                    |
|           | 1   | Torque saturation error protection   | ○         | ○                     |                                    |
| 18        | 0   | Regenerative overload protection   | ○         |                       | ○                                  |
|           | 1   | Regenerative transistor error protection   | ○         |                       |                                    |
| 21        | 0   | Encoder communication line breakage fault protection                             | ○         |                       |                                    |
|           | 1   | Encoder communication error protection   | ○         |                       |                                    |
| 23        | 0   | Encoder communication data error protection                                      | ○         |                       |                                    |
| 24        | 0   | Position deviation excess protection   | ○         | ○                     | ○                                  |
|           | 1   | Speed deviation excess protection  | ○         | ○                     | ○                                  |
| 25        | 0   | Hybrid deviation excess protection   | ○         |                       | ○                                  |
| 26        | 0   | Overspeed protection   | ○         | ○                     | ○                                  |
|           | 1   | 2nd overspeed protection   | ○         | ○                     |                                    |
| 27        | 1   | Absolute clearing protection   | ○         |                       |                                    |
|           | 4   | Command error protection   | ○         |                       | ○                                  |
|           | 5   | Command generation error protection  | ○         |                       | ○                                  |
|           | 6   | Operation command contention protection  | ○         | ○                     |                                    |
|           | 7   | Position information initialization error protection                             | ○         |                       |                                    |

| Error No. |        | Alarm name  | Attribute |                       |                                    |
|-----------|--------|---|-----------|-----------------------|------------------------------------|
| Main      | Sub    |   | History   | Can be cleared        | Emergency stop <sup>(Note 6)</sup> |
| 28        | 0      | Pulse regeneration limit protection                     | ○         | ○                     | ○                                  |
| 29        | 1      | Counter overflow protection 1                           | ○         |                       |                                    |
|           | 2      | Counter overflow protection 2                           | ○         |                       |                                    |
| 31        | 0      | Safety function error protection 1                      | ○         |                       |                                    |
|           | 2      | Safety function error protection 2                      | ○         |                       |                                    |
| 33        | 0      | Input duplicated allocation error-1 protection          | ○         |                       |                                    |
|           | 1      | Input duplicated allocation error-2 protection          | ○         |                       |                                    |
|           | 2      | Input function number error-1 protection                | ○         |                       |                                    |
|           | 3      | Input function number error-2 protection                | ○         |                       |                                    |
|           | 4      | Output function number error-1 protection               | ○         |                       |                                    |
|           | 5      | Output function number error-2 protection               | ○         |                       |                                    |
|           | 8      | Latch input allocation error protection                 | ○         |                       |                                    |
| 34        | 0      | Motor operable range setting error protection           | ○         | ○                     |                                    |
|           | 1      | One revolution absolute operable range error protection | ○         | ○                     |                                    |
| 36        | 0 to 1 | EEPROM parameter error protection                       |           |                       |                                    |
| 37        | 0 to 2 | EEPROM check code error protection                      |           |                       |                                    |
| 38        | 0      | Over-travel inhibit input protection 1                  |           | ○                     |                                    |
|           | 1      | Over-travel inhibit input protection 2                  |           | ○                     |                                    |
|           | 2      | Over-travel inhibit input protection 3                  | ○         |                       |                                    |
| 40        | 0      | Absolute system failure protection                      | ○         | ○ <sup>(Note 2)</sup> |                                    |
| 41        | 0      | Absolute counter limit excess protection                | ○         |                       |                                    |
| 42        | 0      | Absolute overspeed protection                           | ○         | ○ <sup>(Note 2)</sup> |                                    |
| 44        | 0      | Single-turn counter error protection                    | ○         |                       |                                    |
| 45        | 0      | Multi-turn counter error protection                     | ○         |                       |                                    |
| 47        | 0      | Absolute status error protection                        | ○         |                       |                                    |
| 50        | 0      | External scale wiring error protection                  | ○         |                       |                                    |
|           | 1      | External scale communication error protection           | ○         |                       |                                    |
|           | 2      | External scale communication data error protection      | ○         |                       |                                    |
| 51        | 0      | External scale ST error protection 0                    | ○         |                       |                                    |
|           | 1      | External scale ST error protection 1                    | ○         |                       |                                    |
|           | 2      | External scale ST error protection 2                    | ○         |                       |                                    |
|           | 3      | External scale ST error protection 3                    | ○         |                       |                                    |
|           | 4      | External scale ST error protection 4                    | ○         |                       |                                    |
|           | 5      | External scale ST error protection 5                    | ○         |                       |                                    |
| 55        | 0      | Phase-A wiring error protection                         | ○         |                       |                                    |

## 12.2 RTEX communication error

| Error No. |     | Alarm name   | Attribute |                |                                    |
|-----------|-----|--|-----------|----------------|------------------------------------|
| Main      | Sub |  | History   | Can be cleared | Emergency stop <sup>(Note 6)</sup> |
|           | 1   | Phase-B wiring error protection                                | ○         |                |                                    |
|           | 2   | Phase-Z wiring error protection                                | ○         |                |                                    |
| 70        | 0   | Phase U current detector error protection                      | ○         |                |                                    |
|           | 1   | Phase W current detector error protection                      | ○         |                |                                    |
| 72        | 0   | Thermal relay error protection                                 | ○         |                |                                    |
| 80        | 3   | PLL incomplete error protection                                | ○         | ○              |                                    |
| 82        | 0   | RTEX node addressing error protection                          | ○         |                |                                    |
| 83        | 0   | RTEX continuous communication error protection 1               | ○         | ○              | ○                                  |
|           | 1   | RTEX continuous communication error protection 2               | ○         | ○              | ○                                  |
| 84        | 0   | RTEX Communication timeout error protection                    | ○         | ○              | ○                                  |
|           | 3   | RTEX communication synchronization error protection            | ○         |                |                                    |
|           | 5   | RTEX communication cycle error protection                      | ○         | ○              | ○                                  |
| 85        | 0   | Retracting operation completion (I/O) <sup>(Note 7)</sup>      | ○         | (Note 8)       | ○                                  |
|           | 2   | Retracting operation error <sup>(Note 7)</sup>                 | ○         | (Note 8)       | ○                                  |
| 86        | 0   | RTEX cyclic data error protection 1                            | ○         | ○              | ○                                  |
|           | 1   | RTEX cyclic data error protection 2                            | ○         | ○              | ○                                  |
|           | 2   | RTEX update counter error protection                           | ○         |                | ○                                  |
| 87        | 0   | Forced alarm input protection                                  |           | ○              | ○                                  |
|           | 1   | Retracting operation completion (I/O) <sup>(Note 7)</sup>      | ○         | (Note 8)       | ○                                  |
|           | 3   | Retracting operation error <sup>(Note 7)</sup>                 | ○         | (Note 8)       | ○                                  |
| 90        | 2   | RTEX multi-axis synchronization establishment error protection | ○         |                |                                    |
| 91        | 1   | RTEX command error protection                                  | ○         | ○              |                                    |
|           | 3   | RTEX command error protection 2                                | ○         | ○              |                                    |
| 92        | 0   | Encoder data restoration error protection                      | ○         |                |                                    |
|           | 1   | External scale data restoration error protection               | ○         |                |                                    |
|           | 3   | Multi-turn data upper-limit value mismatch error protection    | ○         |                |                                    |
| 93        | 0   | Parameter setting error protection 1                           | ○         |                |                                    |
|           | 2   | Parameter setting error protection 2                           | ○         |                |                                    |
|           | 3   | External scale connection error protection                     | ○         |                |                                    |
|           | 5   | Parameter setting error protection 4                           | ○         |                |                                    |
|           | 8   | Parameter setting error protection 6                           | ○         |                |                                    |
| 94        | 2   | Home return error protection                                   | ○         | ○              |                                    |
|           | 3   | Home return error protection 2                                 | ○         | ○              |                                    |

| Error No.     |        | Alarm name                                   | Attribute |                |                                    |
|---------------|--------|--|-----------|----------------|------------------------------------|
| Main          | Sub    |  | History   | Can be cleared | Emergency stop <sup>(Note 6)</sup> |
| 95            | 0 to 4 | Motor automatic recognition error protection |           |                |                                    |
| 96            | 2      | Control unit error protection 1              | ○         |                |                                    |
|               | 3      | Control unit error protection 2              | ○         |                |                                    |
|               | 4      | Control unit error protection 3              | ○         |                |                                    |
|               | 5      | Control unit error protection 4              | ○         |                |                                    |
|               | 6      | Control unit error protection 5              | ○         |                |                                    |
|               | 7      | Control unit error protection 6              | ○         |                |                                    |
| 98            | 1      | RTEX hardware error protection 1             | ○         |                |                                    |
|               | 2      | RTEX hardware error protection 2             | ○         |                |                                    |
|               | 3      | RTEX hardware error protection 3             | ○         |                |                                    |
| Other numbers |        | Other error protections                      | -         | -              | -                                  |

(Note 1) When Err 16.0 “Over-load protection” occurs, it can be cleared approx. 10 seconds after it occurs. The alarm clear command is received as is and clearing process takes place after it is ready to be cleared.

(Note 2) When Err 40.0 “Absolute system failure protection” or Err 42.0 “Absolute overspeed protection” occurs, the error cannot be cleared until absolute clear is performed.

(Note 3) When an alarm that cannot be cleared occurs, cycle the control power supply after removing the cause of the error or use RTEX software reset command to clear the alarm.

(Note 4) When an alarm that can be cleared occurs, use RTEX communication or USB communication (setup support software) to clear the alarm. Always clear the alarm while all axes are stopped and after securing safety.

(Note 5) If the internal control circuit of the servo amplifier malfunctions due to excessive noise etc., the display will be as shown below.



In such a case, immediately turn OFF the power.

(Note 6) Emergency stop refers to an alarm that is triggered if Pr 5.10 “Sequence at alarm” is set to 4 to 7 and that causes an immediate stop. For details, refer to the instruction manual and other technical references for the servo amplifier.

(Note 7) The alarm generated during retracting operation is switched by Pr 6.86 “Retreat operation alarm setup” bit 15.

Example: When bit 15 = 0, Err 85.0 and Err 85.2 will occur (A5N compatible specification).

When bit 15 = 1, Err 87.1 and Err 87.3 will occur (A6B compatible specification).

(Note 8) Whether alarm can be cleared or not is determined by the setting (bit 0 or 2) of Pr 6.86.

Bit 0: Err 85.0 / Err 87.1 (Retracting operation completion (I/O)) alarm clear attribute

Bit 2: Err 85.2 / Err 87.3 (Retracting operation error) alarm clear attribute; For either case, 0: Alarm clear invalid, 1: Alarm clear valid

## 12.2 RTEX communication error

### 12.2.3 Warning Codes

#### ■ ALARM\_WARNING\_CODES (Structure)

| Member     | Type | Description |
|------------|------|-------------|
| byMainCode | BYTE | Main code   |
| bySubCode  | BYTE | Sub-code    |

#### ■ General warnings

| Warning No.<br>(hexadecimal) | Warning name                                | Description  | Warning latch      | Output setting                 | Warning mask  |
|------------------------------|---|--|--------------------|--------------------------------|---|
|                              |   |  | Pr6.27<br>(Note 1) | Pr4.40 /<br>Pr4.41<br>(Note 2) | Pr6.38 /<br>Pr6.39<br>Corresponding bit<br>(Note 3) |
| A0                           | Overload warning<br>Warning                 | Load factor is 85% or more of the protection level.  | ○                  | 1                              | Pr6.38<br>bit7                                      |
| A1                           | Over-regeneration warning                   | Regenerative load factor has exceeded 85% of the protection level.   | ○                  | 2                              | Pr6.38<br>bit5                                      |
| A2                           | Battery warning<br>(Note 4)                 | Battery voltage is 3.2 V or less.  | Latch fixed        | 3                              | Pr6.38<br>bit0                                      |
| A3                           | Fan warning                                 | Fan has stopped for 1 second.  | ○                  | 4                              | Pr6.38<br>bit6                                      |
| A4                           | Encoder communication warning               | The number of successive encoder communication errors has exceeded the specified value.                        | ○                  | 5                              | Pr6.38<br>bit4                                      |
| A5                           | Encoder overheat warning                    | The encoder temperature exceeds the specified value.   | ○                  | 6                              | Pr6.38<br>bit3                                      |
| A6                           | Oscillation detection warning               | Oscillation state was detected.  | ○                  | 7                              | Pr6.38<br>bit13                                     |
| A7                           | Lifetime detection warning                  | The remaining life expectancy of a capacitor or a fan dropped below the specified value.                       | Latch fixed        | 8                              | Pr6.38<br>bit2                                      |
| A8                           | External scale error warning                | The external scale detected a warning.   | ○                  | 9                              | Pr6.38<br>bit8                                      |
| A9                           | External scale communication warning        | The number of successive external scale communication errors has exceeded the specified value.                 | ○                  | 10                             | Pr6.38<br>bit14                                     |
| AC                           | Deterioration diagnosis warning<br>(Note 6) | Load characteristic estimated value or torque command value at a constant velocity has exceeded the set range. | ○                  | 22                             | Pr6.39<br>bit7                                      |



### ■ Extended warning

| Warning No.<br>(hexadecimal) | Warning name                                 | Description  | Warning latch      | Output setting                 | Warning mask  |
|------------------------------|--|--|--------------------|--------------------------------|---|
|                              |  |  | Pr6.27<br>(Note 1) | Pr4.40 /<br>Pr4.41<br>(Note 2) | Pr6.38 /<br>Pr6.39<br>Corresponding bit<br>(Note 3) |
| C0                           | RTEX continuous communication error warning  | The number of successive errors (CRC error) detected when reading the received data sent to the local node<br>The number of successive errors (CRC error) has exceeded the value set by Pr 7.26 "RTEX continuous communication error warning setup". | ○                  | 11                             | Pr6.38 bit9   |
| C1                           | RTEX accumulated communication error warning | The number of successive errors (CRC error) detected when reading the received data sent to the local node<br>has exceeded the value set by Pr 7.27 "RTEX accumulated communication error warning setup".  | Latch fixed        | 12                             | Pr6.38 bit10  |
| C2                           | RTEX_Update_Counter error warning            | The Update_Counter was not updated properly because the data accumulated exceeded the count value set by Pr 7.28 "RTEX_Update_Counter error warning setup".  | Latch fixed        | 13                             | Pr6.38 bit11  |
| C3                           | Main power OFF warning                       | When Pr 7.14 "Main power OFF warning detection time" was set to 10 to 1999, instantaneous power failure that occurred between L1 and L3 exceeded the time set by Pr 7.14.  | ○                  | 14                             | Pr6.38 bit12  |
| D2                           | PANATERM command execution warning           | While bit 0 of Pr 7.99 "RTEX function enhancement setting 6" was set to 1 and RTEX communication was established, an operation command (test run, FFT, etc.) was executed by the setup support software "PANATERM".                                  | ○                  | 30                             | Pr6.39 bit8   |

(Note 1) The symbol "○" marked in the "Warning latch" column indicates that it is possible to switch the mode between non-latch mode (latch for 1 second) and latch mode by using Pr 6.27 "Warning latch state setup". Only latch mode is available for the battery warning and the lifetime detection warning.

(Note 2) Select the warning output signal 1 (WARN 1) or warning output signal 2 (WARN 2) through Pr 4.40 "Warning output select 1" or Pr 4.41 "Warning output select 2". When the set value is 0, all warnings are Ored before being output. Do not use any settings other than the settings shown in the above table.

(Note 3) Each warning detection can be disabled by Pr 6.38 "Warning mask setting" or Pr 6.39 "Warning mask setting 2". The corresponding bits are shown in the table. Set the bit to 1 to disable the warning detection. For extended warning, warning detection can be disabled by parameter settings.

Also note that bit arrangements of these masks are different from those of general-purpose type MINAS-A6 series.

## 12.2 RTEX communication error

---

- (Note 4) When the single-turn absolute function is enabled, a battery alarm is not detected.
- (Note 5) Warning can be cleared by alarm clear. If warning cause is not resolved yet, the warning is cleared once, but a warning is issued again.
- (Note 6) If bit 1 of Pr 6.97 "Function enhancement setup 3" is set to 0, it is disabled.

## 12.3 List of AMP Parameters

### 12.3.1 Class 0: Basic Setting

| Class | No.   | Parameter name   | Unit         | Setting range |
|-------|---|--|--------------|---------------|
| 0     | 00  | Rotational direction setup                                   | -            | 0 to 1        |
|       | 01  | Control mode setup   | -            | 0 to 6        |
|       | 02  | Real-time auto-gain tuning setup                             | -            | 0 to 6        |
|       | 03  | Selection of machine stiffness at real-time auto-gain tuning | -            | 0 to 31       |
|       | 04  | Inertia ratio  | %            | 0 to 10000    |
|       | 08  | Command pulse counts per one motor revolution                | pulse        | 0 to $2^{23}$ |
|       | 09  | Numerator of electronic gear                                 | -            | 0 to $2^{30}$ |
|       | 10  | Denominator of electronic gear                               | -            | 1 to $2^{30}$ |
|       | 11  | Number of output pulses per motor rotation                   | pulse/r      | 1 to 2097152  |
|       | 12  | Reversal of pulse output logic / output source selection     | -            | 0 to 3        |
|       | 13  | 1st torque limit   | %            | 0 to 500      |
|       | 14  | Position deviation excess setup                              | Command unit | 0 to $2^{30}$ |
|       | 15  | Absolute encoder setup                                       | -            | 0 to 4        |
|       | 16  | External regenerative resistor setup                         | -            | 0 to 3        |
| 17    | Load factor of external regenerative resistor selection | -  | 0 to 4       |               |

### 12.3.2 Class 1: Gain Adjustment

| Class | No.                           | Parameter name                                 | Unit    | Setting range |
|-------|-------------------------------|--|---------|---------------|
| 1     | 00                            | 1st gain of position loop                      | 0.1/s   | 0 to 30000    |
|       | 01                            | 1st gain of velocity loop                      | 0.1 Hz  | 1 to 32767    |
|       | 02                            | 1st time constant of velocity loop integration | 0.1 ms  | 1 to 10000    |
|       | 03                            | 1st filter of speed detection                  | -       | 0 to 5        |
|       | 04                            | 1st time constant of torque filter             | 0.01 ms | 0 to 2500     |
|       | 05                            | 2nd gain of position loop                      | 0.1/s   | 0 to 30000    |
|       | 06                            | 2nd gain of velocity loop                      | 0.1 Hz  | 1 to 32767    |
|       | 07                            | 2nd time constant of velocity loop integration | 0.1 ms  | 1 to 10000    |
| 08    | 2nd filter of speed detection | -  | 0 to 5  |               |

## 12.3 List of AMP Parameters

| Class | No.                                    | Parameter name                           | Unit       | Setting range |
|-------|--|--|------------|---------------|
|       | 09                                     | 2nd time constant of torque filter       | 0.01 ms    | 0 to 2500     |
|       | 10                                     | Velocity feed forward gain               | 0.1%       | 0 to 4000     |
|       | 11                                     | Velocity feed forward gain               | 0.1%       | 0 to 4000     |
|       | 12                                     | Velocity feed forward gain               | 0.01 ms    | 0 to 6400     |
|       | 13                                     | Torque feed forward filter               | 0.01 ms    | 0 to 6400     |
|       | 14                                     | 2nd gain setup                           | -          | 0 to 1        |
|       | 15                                     | Mode of position control switching       | -          | 0 to 10       |
|       | 16                                     | Delay time of position control switching | 0.1 ms     | 0 to 10000    |
|       | 17                                     | Level of position control switching      | -          | 0 to 20000    |
|       | 18                                     | Hysteresis at position control switching | -          | 0 to 20000    |
|       | 19                                     | Position gain switching time             | 0.1 ms     | 0 to 10000    |
|       | 20                                     | Mode of velocity control switching       | -          | 0 to 5        |
|       | 21                                     | Delay time of velocity control switching | 0.1 ms     | 0 to 10000    |
|       | 22                                     | Level of velocity control switching      | -          | 0 to 20000    |
|       | 23                                     | Hysteresis at velocity control switching | -          | 0 to 20000    |
|       | 24                                     | Mode of torque control switching         | -          | 0 to 3        |
|       | 25                                     | Delay time of torque control switching   | 0.1 ms     | 0 to 10000    |
| 26    | Level of torque control switching      | -  | 0 to 20000 |               |
| 27    | Hysteresis at torque control switching | -  | 0 to 20000 |               |

### 12.3.3 Class 2: Vibration Suppression Function

| Class | No. | Parameter name             | Unit | Setting range |
|-------|-----|----------------------------|------|---------------|
| 2     | 00  | Adaptive filter mode setup | -    | 0 to 6        |
|       | 01  | 1st notch frequency        | Hz   | 50 to 5000    |
|       | 02  | 1st notch width selection  | -    | 0 to 20       |
|       | 03  | 1st notch depth selection  | -    | 0 to 99       |
|       | 04  | 2nd notch frequency        | Hz   | 50 to 5000    |
|       | 05  | 2nd notch width selection  | -    | 0 to 20       |
|       | 06  | 2nd notch depth selection  | -    | 0 to 99       |
|       | 07  | 3rd notch frequency        | Hz   | 50 to 5000    |
|       | 08  | 3rd notch width selection  | -    | 0 to 20       |
|       | 09  | 3rd notch depth selection  | -    | 0 to 99       |

## 12.3 List of AMP Parameters

| Class | No.                                 | Parameter name                        | Unit      | Setting range |
|-------|-------------------------------------|---------------------------------------|-----------|---------------|
|       | 10                                  | 4th notch frequency                   | Hz        | 50 to 5000    |
|       | 11                                  | 4th notch width selection             | -         | 0 to 20       |
|       | 12                                  | 4th notch depth selection             | -         | 0 to 99       |
|       | 13                                  | Selection of damping filter switching | -         | 0 to 6        |
|       | 14                                  | 1st damping frequency                 | 0.1 Hz    | 0 to 3000     |
|       | 15                                  | 1st damping filter setup              | 0.1 Hz    | 0 to 1500     |
|       | 16                                  | 2nd damping frequency                 | 0.1 Hz    | 0 to 3000     |
|       | 17                                  | 2nd damping filter setup              | 0.1 Hz    | 0 to 1500     |
|       | 18                                  | 3rd damping frequency                 | 0.1 Hz    | 0 to 3000     |
|       | 19                                  | 3rd damping filter setup              | 0.1 Hz    | 0 to 1500     |
|       | 20                                  | 4th damping frequency                 | 0.1 Hz    | 0 to 3000     |
|       | 21                                  | 4th damping filter setup              | 0.1 Hz    | 0 to 1500     |
|       | 22                                  | Command smoothing filter              | 0.1 ms    | 0 to 10000    |
|       | 23                                  | Command FIR filter                    | 0.1 ms    | 0 to 10000    |
|       | 24                                  | 5th notch frequency                   | Hz        | 50 to 5000    |
|       | 25                                  | 5th notch width selection             | -         | 0 to 20       |
|       | 26                                  | 5th notch depth selection             | -         | 0 to 99       |
|       | 27                                  | 1st vibration control width setting   | -         | 0 to 1000     |
|       | 28                                  | 2nd vibration control width setting   | -         | 0 to 1000     |
|       | 29                                  | 3rd vibration control width setting   | -         | 0 to 1000     |
| 30    | 4th vibration control width setting | -                                     | 0 to 1000 |               |

### 12.3.4 Class 3: Speed, Torque Control, Full-closed Control

| Class | No. | Parameter name                                   | Unit            | Setting range        |
|-------|-----|--|-----------------|----------------------|
| 3     | 12  | Acceleration time setting                        | ms/(1000 r/min) | 0 to 10000           |
|       | 13  | Deceleration time setting                        | ms/(1000 r/min) | 0 to 10000           |
|       | 14  | Sigmoid acceleration/<br>deceleration time setup | ms              | 0 to 10000           |
|       | 17  | Speed limit selection                            | -               | 0 to 1               |
|       | 21  | Speed limit value 1                              | r/min           | 0 to 20000           |
|       | 22  | Speed limit value 2                              | r/min           | 0 to 20000           |
|       | 23  | External scale selection                         | -               | 0 to 6               |
|       | 24  | External scale numerator<br>of division          | -               | 0 to 2 <sup>23</sup> |
|       | 25  | External scale<br>denominator of division        | -               | 1 to 2 <sup>23</sup> |

## 12.3 List of AMP Parameters

| Class | No. | Parameter name   | Unit                | Setting range        |
|-------|-----|--|---------------------|----------------------|
|       | 26  | External scale reversal of direction   | -                   | 0 to 3               |
|       | 27  | External scale Z phase disconnection detection disable                         | -                   | 0 to 1               |
|       | 28  | Hybrid deviation excess protection   | Command unit        | 1 to 2 <sup>27</sup> |
|       | 29  | Hybrid deviation clear setting   | Rotation            | 0 to 100             |
|       | 32  | External scale movement judgment threshold at virtual full-closed control mode | External scale unit | 0 to 65534           |

### 12.3.5 Class 4: I/O Monitor Setting

| Class | No.                      | Parameter name                                  | Unit         | Setting range  |
|-------|--------------------------|---|--------------|----------------|
| 4     | 00                       | SI1 input selection                             | -            | 0 to 00FFFFFFh |
|       | 01                       | SI2 input selection                             | -            | 0 to 00FFFFFFh |
|       | 02                       | SI3 input selection                             | -            | 0 to 00FFFFFFh |
|       | 03                       | SI4 input selection                             | -            | 0 to 00FFFFFFh |
|       | 04                       | SI5 input selection                             | -            | 0 to 00FFFFFFh |
|       | 05                       | SI6 input selection                             | -            | 0 to 00FFFFFFh |
|       | 06                       | SI7 input selection                             | -            | 0 to 00FFFFFFh |
|       | 07                       | SI8 input selection                             | -            | 0 to 00FFFFFFh |
|       | 10                       | SO1 output selection                            | -            | 0 to 00FFFFFFh |
|       | 11                       | SO2 output selection                            | -            | 0 to 00FFFFFFh |
|       | 12                       | SO3 output selection                            | -            | 0 to 00FFFFFFh |
|       | 16                       | Type of analog monitor 1                        | -            | 0 to 28        |
|       | 17                       | Analog monitor 1 output gain                    | -            | 0 to 214748364 |
|       | 18                       | Type of analog monitor 2                        | -            | 0 to 28        |
|       | 19                       | Analog monitor 2 output gain                    | -            | 0 to 214748364 |
|       | 21                       | Analog monitor output setup                     | -            | 0 to 2         |
|       | 31                       | Positioning complete range                      | Command unit | 0 to 2097152   |
|       | 32                       | Positioning complete (In-position) output setup | -            | 0 to 10        |
|       | 33                       | INP hold time                                   | ms           | 0 to 30000     |
|       | 34                       | Zero-speed                                      | r/min        | 10 to 20000    |
| 35    | Speed coincidence range  | r/min   | 10 to 20000  |                |
| 36    | At-speed (Speed arrival) | r/min   | 10 to 20000  |                |

## 12.3 List of AMP Parameters

| Class | No. | Parameter name                                    | Unit         | Setting range             |
|-------|-----|---|--------------|---------------------------|
|       | 37  | Mechanical brake action at stalling setup         | ms           | 0 to 10000                |
|       | 38  | Mechanical brake action at running setup          | ms           | 0 to 32000                |
|       | 39  | Brake release speed setup                         | r/min        | 30 to 3000                |
|       | 40  | Selection of alarm output 1                       | -            | 0 to 40                   |
|       | 41  | Selection of alarm output 2                       | -            | 0 to 40                   |
|       | 42  | 2nd Positioning complete (In-position) range      | Command unit | 0 to 2097152              |
|       | 44  | Position compare output pulse width setting       | 0.1 ms       | 0 to 32767                |
|       | 45  | Position compare output polarity select           | -            | 0 to 7                    |
|       | 47  | Pulse output select                               | -            | 0 to 1                    |
|       | 48  | Position compare value 1                          | Command unit | -2147483648 to 2147483647 |
|       | 49  | Position compare value 2                          | Command unit | -2147483648 to 2147483647 |
|       | 50  | Position compare value 3                          | Command unit | -2147483648 to 2147483647 |
|       | 51  | Position compare value 4                          | Command unit | -2147483648 to 2147483647 |
|       | 52  | Position compare value 5                          | Command unit | -2147483648 to 2147483647 |
|       | 53  | Position compare value 6                          | Command unit | -2147483648 to 2147483647 |
|       | 54  | Position compare value 7                          | Command unit | -2147483648 to 2147483647 |
|       | 55  | Position compare value 8                          | Command unit | -2147483648 to 2147483647 |
|       | 56  | Position compare output delay compensation amount | 0.1 us       | -32768 to 32767           |
|       | 57  | Position compare output assignment setting        | -            | -2147483648 to 2147483647 |

### 12.3.6 Class 5: Enhancing Setting

| Class | No. | Parameter name                       | Unit | Setting range |
|-------|-----|--------------------------------------|------|---------------|
| 5     | 03  | Denominator of pulse output division | -    | 0 to 8388608  |
|       | 04  | Over-travel inhibit input setup      | -    | 0 to 2        |
|       | 05  | Sequence at over-travel inhibit      | -    | 0 to 2        |

## 12.3 List of AMP Parameters

| Class | No. | Parameter name  | Unit            | Setting range   |
|-------|-----|---|-----------------|---|
|       | 06  | Sequence at Servo-Off                                     | -               | 0 to 9  |
|       | 07  | Sequence at main power OFF                                | -               | 0 to 9  |
|       | 08  | LV trip selection at main power OFF                       | -               | 0 to 3  |
|       | 09  | Detection time of main power off                          | ms              | 20 to 2000" <a href="#">12.3.9 Class 8: Special Setting 3</a> " |
|       | 10  | Sequence at alarm   | -               | 0 to 7  |
|       | 11  | Torque setup for emergency stop                           | %               | 0 to 500  |
|       | 12  | Over-load level setup                                     | %               | 0 to 500  |
|       | 13  | Over-speed level setup                                    | r/min           | 0 to 20000  |
|       | 14  | Motor working range setup                                 | 0.1 revolution  | 0 to 1000   |
|       | 15  | Control input signal read setting                         | -               | 0 to 3  |
|       | 20  | Position setup unit select                                | -               | 0 to 1  |
|       | 21  | Selection of torque limit                                 | -               | 0 to 4  |
|       | 22  | 2nd torque limit  | %               | 0 to 500  |
|       | 23  | Torque limit switching setup 1                            | ms/100 %        | 0 to 4000   |
|       | 24  | Torque limit switching setup 2                            | ms/100 %        | 0 to 4000   |
|       | 25  | Positive direction torque limit                           | %               | 0 to 500  |
|       | 26  | Negative direction torque limit                           | %               | 0 to 500  |
|       | 31  | USB axis address  | -               | 0 to 127  |
|       | 33  | Pulse regenerative output limit setup                     | -               | 0 to 1  |
|       | 45  | Quadrant projection positive direction compensation value | 0.1%            | -1000 to 1000   |
|       | 46  | Quadrant projection negative direction compensation value | 0.1%            | -1000 to 1000   |
|       | 47  | Quadrant projection compensation delay time               | ms              | 0 to 1000   |
|       | 48  | Quadrant projection compensation filter setting L         | 0.01 ms         | 0 to 6400   |
|       | 49  | Quadrant projection compensation filter setting H         | 0.1 ms          | 0 to 10000  |
|       | 56  | Slow stop deceleration time setting                       | ms/(1000 r/min) | 0 to 10000  |



## 12.3 List of AMP Parameters

| Class | No. | Parameter name  | Unit                | Setting range   |
|-------|-----|---|---------------------|-----------------|
|       | 57  | Slow stop S-shape acceleration and deceleration setting | ms                  | 0 to 1000       |
|       | 66  | Deterioration diagnosis convergence judgment time       | 0.1 s               | 0 to 10000      |
|       | 67  | Deterioration diagnosis inertia ratio upper limit       | %                   | 0 to 10000      |
|       | 68  | Deterioration diagnosis inertia ratio lower limit       | %                   | 0 to 10000      |
|       | 69  | Deterioration diagnosis unbalanced load upper limit     | 0.1%                | -1000 to 1000   |
|       | 70  | Deterioration diagnosis unbalanced load lower limit     | 0.1%                | -1000 to 1000   |
|       | 71  | Deterioration diagnosis dynamic friction upper limit    | 0.1%                | -1000 to 1000   |
|       | 72  | Deterioration diagnosis dynamic friction lower limit    | 0.1%                | -1000 to 1000   |
|       | 73  | Deterioration diagnosis viscous friction upper limit    | 0.1%/ (10000 r/min) | 0 to 10000      |
|       | 74  | Deterioration diagnosis viscous friction lower limit    | 0.1%/ (10000 r/min) | 0 to 10000      |
|       | 75  | Deterioration diagnosis velocity setting                | r/min               | -20000 to 20000 |
|       | 76  | Deterioration diagnosis torque average time             | ms                  | 0 to 10000      |
|       | 77  | Deterioration diagnosis torque upper limit              | 0.1%                | -1000 to 1000   |
|       | 78  | Deterioration diagnosis torque lower limit              | 0.1%                | -1000 to 1000   |

(Note 1) When using this setup value at a value smaller than the default value, confirm that it matches the user's power supply environment.

### 12.3.7 Class 6: Special Setting 1

| Class | No. | Parameter name                           | Unit   | Setting range |
|-------|-----|--|--------|---------------|
| 6     | 02  | Speed deviation excess setup             | r/min  | 0 to 20000    |
|       | 05  | Position control 3rd gain effective time | 0.1 ms | 0 to 10000    |
|       | 06  | Position control 3rd gain scale factor   | %      | 50 to 1000    |
|       | 07  | Additional value to torque command       | %      | -100 to 100   |

## 12.3 List of AMP Parameters

| Class | No. | Parameter name   | Unit                | Setting range   |
|-------|-----|--|---------------------|-----------------|
|       | 08  | Torque compensation value in positive direction                  | %                   | -100 to 100     |
|       | 09  | Torque compensation value in negative direction                  | %                   | -100 to 100     |
|       | 10  | Function expansion setup   | -                   | -32768 to 32767 |
|       | 11  | Current response setup   | %                   | 10 to 100       |
|       | 14  | Immediate stop time at the time of alarming                      | ms                  | 1000            |
|       | 15  | 2nd over-speed level setup                                       | r/min               | 0 to 20000      |
|       | 18  | Power turn-on wait time  | 0.1 s               | 0 to 100        |
|       | 22  | A, B phase external scale pulse output method selection          | -                   | 0 to 1          |
|       | 23  | Load fluctuation correction gain                                 | %                   | -100 to 100     |
|       | 24  | Load fluctuation correction filter                               | 0.01 ms             | 10 to 2500      |
|       | 27  | Alarm latch time selection                                       | -                   | 0 to 3          |
|       | 31  | Real time auto tuning estimation speed                           | -                   | 0 to 3          |
|       | 32  | Real time auto tuning custom setup                               | -                   | -32768 to 32767 |
|       | 34  | Hybrid vibration suppression gain                                | -                   | 0 to 30000      |
|       | 35  | Hybrid vibration suppression filter                              | 0.1/s               | 0 to 32000      |
|       | 36  | Dynamic brake operation input setup                              | 0.01 ms             | 0 to 1          |
|       | 37  | Oscillation detecting level                                      | -                   | 0 to 1000       |
|       | 38  | Alarm mask setup   | 0.1%                | -32768 to 32767 |
|       | 39  | Alarm mask setup 2   | -                   | -32768 to 32767 |
|       | 41  | 1st damping depth  | -                   | 0 to 1000       |
|       | 42  | Two-stage torque filter time constant                            | -                   | 0 to 2500       |
|       | 43  | Two-stage torque filter damping term                             | 0.01 ms             | 0 to 1000       |
|       | 47  | Function expansion settings 2                                    | -32768 to 32767     | -32768 to 32767 |
|       | 48  | Adjustment filter  | 0 to 2000           | 0 to 2000       |
|       | 49  | Command response filter / adjustment filter damping term setting | 0 to 99             | 0 to 99         |
|       | 50  | Viscous friction compensation gain                               | 0.1%/ (10000 r/min) | 0 to 10000      |
|       | 51  | Immediate stop completion wait time                              | ms                  | 0 to 10000      |
|       | 57  | Torque saturation error protection detection time                | ms                  | 0 to 5000       |
|       | 60  | 2nd damping depth  | -                   | 0 to 1000       |
|       | 61  | 1st resonance frequency  | 0.1 Hz              | 0 to 3000       |

## 12.3 List of AMP Parameters

| Class | No.                          | Parameter name                           | Unit                      | Setting range   |
|-------|------------------------------|--|---------------------------|-----------------|
|       | 62                           | 1st resonance damping ratio              | -                         | 0 to 1000       |
|       | 63                           | 1st anti-resonance frequency             | 0.1 Hz                    | 0 to 3000       |
|       | 64                           | 1st anti-resonance damping ratio         | -                         | 0 to 1000       |
|       | 65                           | 1st response frequency                   | 0.1 Hz                    | 0 to 3000       |
|       | 66                           | 2nd resonance frequency                  | 0.1 Hz                    | 0 to 3000       |
|       | 67                           | 2nd resonance damping ratio              | -                         | 0 to 1000       |
|       | 68                           | 2nd anti-resonance frequency             | 0.1 Hz                    | 0 to 3000       |
|       | 69                           | 2nd anti-resonance damping ratio         | -                         | 0 to 1000       |
|       | 70                           | 2nd response frequency                   | 0.1 Hz                    | 0 to 3000       |
|       | 71                           | 3rd damping depth                        | -                         | 0 to 1000       |
|       | 72                           | 4th damping depth                        | -                         | 0 to 1000       |
|       | 73                           | Load estimation filter                   | 0.01 ms                   | 0 to 2500       |
|       | 74                           | Torque compensation frequency 1          | 0.1 Hz                    | 0 to 5000       |
|       | 75                           | Torque compensation frequency 2          | 0.1 Hz                    | 0 to 5000       |
|       | 76                           | Load estimation count                    | -                         | 0 to 8          |
|       | 85                           | Retracting operation condition setting   | -                         | -32768 to 32767 |
|       | 86                           | Retracting operation alarm setting       | -                         | -32768 to 32767 |
|       | 88                           | Absolute multi-rotation data upper limit | -                         | 0 to 65534      |
| 97    | Function expansion setting 3 | -  | -2147483648 to 2147483647 |                 |
| 98    | Function expansion setting 4 | -  | -2147483648 to 2147483647 |                 |

### 12.3.8 Class 7: Special Setting 2

| Class | No. | Parameter name                     | Unit         | Setting range             |
|-------|-----|------------------------------------|--------------|---------------------------|
| 7     | 00  | Display on LED                     | -            | 0 to 32767                |
|       | 01  | Display time setup upon power-up   | 100 ms       | -1 to 1000                |
|       | 03  | Output setup during torque limit   | -            | 0 to 1                    |
|       | 09  | Correction time of latch delay 1   | 25 ns        | -2000 to 2000             |
|       | 10  | Soft limit function                | -            | 0 to 3                    |
|       | 11  | Positive side software limit value | Command unit | -1073741823 to 1073741823 |
|       | 12  | Negative side software limit value | Command unit | -1073741823 to 1073741823 |
|       | 13  | Absolute home position offset      | Command unit | -1073741823 to 1073741823 |

## 12.3 List of AMP Parameters

| Class | No. | Parameter name  | Unit         | Setting range   |
|-------|-----|---|--------------|-----------------|
|       | 14  | Main power OFF warning detection time                                 | ms           | 0 to 2000       |
|       | 15  | Positioning adjacent range  | Command unit | 0 to 1073741823 |
|       | 16  | Torque saturation error protection frequency                          | No. of times | 0 to 30000      |
|       | 20  | RTEX communication cycle setup  | -            | -1 to 12        |
|       | 21  | RTEX command updating cycle ratio setting                             | -            | 1 to 2          |
|       | 22  | RTEX function extended setup 1  | -            | -32768 to 32767 |
|       | 23  | RTEX function extended setup 2  | -            | -32768 to 32767 |
|       | 24  | RTEX function extended setup 3  | -            | -32768 to 32767 |
|       | 25  | RTEX speed unit setup   | -            | 0 to 1          |
|       | 26  | RTEX continuous error warning setup                                   | No. of times | 0 to 32767      |
|       | 27  | RTEX accumulated error warning setup                                  | No. of times | 0 to 32767      |
|       | 28  | RTEX_Update_Counter error warning setup                               | No. of times | 0 to 32767      |
|       | 29  | RTEX monitor select 1   | -            | 0 to 32767      |
|       | 30  | RTEX monitor select 2   | -            | 0 to 32767      |
|       | 31  | RTEX monitor select 3   | -            | 0 to 32767      |
|       | 32  | RTEX monitor select 4   | -            | 0 to 32767      |
|       | 33  | RTEX monitor select 5   | -            | 0 to 32767      |
|       | 34  | RTEX monitor select 6   | -            | 0 to 32767      |
|       | 35  | RTEX command setting 1  | -            | 0 to 2          |
|       | 36  | RTEX command setting 2  | -            | 0 to 2          |
|       | 37  | RTEX command setting 3  | -            | 0 to 2          |
|       | 38  | RTEX_Update_Counter error protection setup                            | No. of times | 0 to 32767      |
|       | 41  | RTEX function extended setup 5  | -            | -32768 to 32767 |
|       | 78  | Signal reading setting for latch trigger with stop function           | -            | 0 to 3          |
|       | 91  | RTEX communication cycle extended setup                               | ns           | 0 to 2000000    |
|       | 92  | Correction time of latch delay 2                                      | 25 ns        | -2000 to 2000   |
|       | 93  | Home return limit speed   | r/min        | 0 to 20000      |
|       | 95  | Number of RTEX continuous communication error protection 1 detections | No. of times | 0 to 17         |
|       | 96  | Number of RTEX continuous communication error protection 2 detections | No. of times | 0 to 17         |

## 12.3 List of AMP Parameters

| Class | No. | Parameter name   | Unit         | Setting range             |
|-------|-----|--|--------------|---------------------------|
|       | 97  | Number of RTEX communication timeout error protection detections   | No. of times | 0 to 17                   |
|       | 98  | Number of RTEX cyclic data error protection 1 / 2 detections       | No. of times | 0 to 17                   |
|       | 99  | RTEX function extended setup 6                                     | -            | -32768 to 32767           |
|       | 108 | RTEX communication synchronization setup                           | -            | 0 to 7                    |
|       | 110 | RTEX function extended setup 7                                     | -            | -2147483648 to 2147483647 |
|       | 111 | Trigger signal allocation setting of latch mode with stop function | -            | 0 to 64                   |
|       | 112 | Selection of RTEX communication status flag                        | -            | 0 to 1                    |

### 12.3.9 Class 8: Special Setting 3

| Class | No. | Parameter name  | Unit                              | Setting range             |
|-------|-----|---|-----------------------------------|---------------------------|
| 8     | 01  | Profile linear acceleration constant                    | 10000 Command unit/s <sup>2</sup> | 1 to 429496               |
|       | 04  | Profile linear deceleration constant                    | 10000 Command unit/s <sup>2</sup> | 1 to 429496               |
|       | 10  | Amount of travel after profile position latch detection | Command unit                      | -1073741823 to 1073741823 |
|       | 12  | Profile home return position mode setup                 | -                                 | 0 to 1                    |
|       | 13  | Profile home return velocity 1                          | Command unit/s or r/min           | 0 to 2147483647           |
|       | 14  | Profile home return velocity 2                          | Command unit/s or r/min           | 0 to 2147483647           |
|       | 17  | Relative movement of retracting operation               | Command unit                      | -2147483648 to 2147483647 |
|       | 18  | Retracting operation speed                              | Command unit/s or r/min           | 0 to 2147483647           |

## 12.4 Monitor Commands

### 12.4 Monitor Commands

These commands are specified with RTEX\_ReadAmpData (amplifier monitor).

| Type_Code<br>(Note 1) (Note 3) |  | Name  |      | Index<br>(Note 2) | Unit                                   | Description  |                |   |   |   |   |  |
|--------------------------------|--|---|------|-------------------|--|--|----------------|---|---|---|---|--|
| A4N<br>comp<br>atible          | Stand<br>ard                                   |   |      |                   |  |  |                |   |   |   |   |  |
| 101h                           | 01h  | Position deviation                                    | PERR | 0<br>(1,2)        | Command<br>unit                        | <p>&lt;In position control mode&gt;<br/>Position deviation<br/>&lt;In full-closed control mode&gt;<br/>External scale deviation<br/>* The computation method (reference) of position deviation and external scale deviation is set in bit 14 of Pr 7.23 "Command position deviation output switching".</p> <table border="1"> <thead> <tr> <th>r7.23<br/>bit14</th> <th>Computation method of<br/>positional deviation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Deviation from the<br/>command after filtering</td> </tr> <tr> <td>1</td> <td>Deviation from the<br/>command before filtering</td> </tr> </tbody> </table> <p>&lt;In speed / torque control mode&gt;<br/>Undefined<br/>Note: Although the same data is returned whether Index is 1 or 2, use Index = 0.</p> | r7.23<br>bit14 | Computation method of<br>positional deviation | 0 | Deviation from the<br>command after filtering | 1 | Deviation from the<br>command before filtering |
| r7.23<br>bit14                 | Computation method of<br>positional deviation  |   |      |                   |  |  |                |   |   |   |   |  |
| 0                              | Deviation from the<br>command after filtering  |   |      |                   |  |  |                |   |   |   |   |  |
| 1                              | Deviation from the<br>command before filtering |   |      |                   |  |  |                |   |   |   |   |  |
| 102h                           | 02h  | Encoder<br>resolution                                 | -    | 0                 | pulse/r                                | Encoder resolution of the motor<br>connected   |                |   |   |   |   |  |
| 104h                           | 04h  | Command<br>position<br>(after filtering)              | MPOS | 0                 | Command<br>unit                        | Command position (after filtering)   |                |   |   |   |   |  |
| 105h                           | 05h  | Actual speed  | ASPD | 0                 | Set the<br>unit<br>through<br>Pr 7.25. | <p>Motor actual speed<br/>* Set the unit through Pr 7.25 "RTEX speed unit setup".</p> <table border="1"> <thead> <tr> <th>Pr7.25</th> <th>Unit</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>[r/min]</td> </tr> <tr> <td>1</td> <td>[Command unit/s]</td> </tr> </tbody> </table>  | Pr7.25         | Unit  | 0 | [r/min]                                       | 1 | [Command unit/s]                               |
| Pr7.25                         | Unit   |   |      |                   |  |  |                |   |   |   |   |  |
| 0                              | [r/min]  |   |      |                   |  |  |                |   |   |   |   |  |
| 1                              | [Command unit/s]                               |   |      |                   |  |  |                |   |   |   |   |  |
| 106h                           | 06h  | Internal<br>command torque                            | TRQ  | 0                 | 0.1%                                   | Command torque to motor  |                |   |   |   |   |  |
| -                              | 07h  | Actual position                                       | APOS | 0                 | Command<br>unit                        | <p>Motor actual position<br/>* Position of the external scale in full-closed mode</p>  |                |   |   |   |   |  |
| -                              | 08h  | Internal<br>command<br>position (before<br>filtering) | IPOS | 0                 | Command<br>unit                        | Internal command position before filtering   |                |   |   |   |   |  |

| Type_Code<br>(Note 1) (Note 3) |                  | Name                                  |       | Index<br>(Note 2) | Unit                          | Description   |        |      |   |         |   |                  |
|--------------------------------|------------------|---------------------------------------|-------|-------------------|-------------------------------|---|--------|------|---|---------|---|------------------|
| A4N compatible                 | Standard         |                                       |       |                   |                               |   |        |      |   |         |   |                  |
| -                              | 09h              | Latch position 1                      | LPOS1 | 0                 | Command unit                  | Motor actual position latched in CH1  |        |      |   |         |   |                  |
| -                              | 0Ah              | Latch position 2                      | LPOS2 | 0                 | Command unit                  | Motor actual position latched in CH2  |        |      |   |         |   |                  |
| -                              | 0Ch              | Command velocity<br>(after filtering) | MSPD  | 0                 | Set the unit through Pr 7.25. | Command velocity after filtering<br>* Set the unit through Pr 7.25 "RTEX speed unit setup".<br><table border="1" data-bbox="852 620 1249 741"> <thead> <tr> <th>Pr7.25</th> <th>Unit</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>[r/min]</td> </tr> <tr> <td>1</td> <td>[Command unit/s]</td> </tr> </tbody> </table><br>* The value is undefined in torque control mode. | Pr7.25 | Unit | 0 | [r/min] | 1 | [Command unit/s] |
| Pr7.25                         | Unit             |                                       |       |                   |                               |   |        |      |   |         |   |                  |
| 0                              | [r/min]          |                                       |       |                   |                               |   |        |      |   |         |   |                  |
| 1                              | [Command unit/s] |                                       |       |                   |                               |   |        |      |   |         |   |                  |
| -                              | 0Dh              | External scale position (Note 4)      | EXPOS | 0                 | Pulse (External scale)        | External scale position   |        |      |   |         |   |                  |

(Note 1) When a Type\_Code error occurs, command error (0031h) will be returned.

Manufacturer will use a Type\_Code not listed above.

When a Type\_Code used by the manufacturer is set, undefined value will be returned in place of command error (0031h).

(Note 2) When an Index error occurs, command error (0032h) will be returned.

(Note 3) A4N compatible: Type\_Code compatible with A4N series can be used, but only with main commands.

Standard: Type\_Code newly created for A5N and A5N series and can be used with both main commands and subcommands. When using with main commands, set leftmost 4 bits to 0.

\* Although the product supports the A4N compatible Type\_Code to maintain compatibility, basically use the standard Type\_Code.

(Note 4) The version before the function extended version 1 is not supported.

| Type_Code      |          | Name                    |   | Index | Unit          | Description   |
|----------------|----------|-------------------------|---|-------|---------------|---|
| A4N compatible | Standard |                         |   |       |               |   |
| 111h           | 11h      | Regenerative load ratio | - | 0     | %<br>(Note 2) | Ratio of the regenerative overload protection to the alarm occurrence level |
| 112h           | 12h      | Overload ratio          | - | 0     | 0.1%          | Ratio of the actual load to the rated motor load                            |
| -              | 21h      | Logical input signal    | - | 0     | -             | Logic level of input signal   |
| -              | 22h      | Logical output signal   | - | 0     | -             | Logic level of output signal  |
| -              | 23h      | Logical input signal    | - | 0     | -             | Logic level of input signal (expansion portion)                             |

## 12.4 Monitor Commands

| Type_Code      |            | Name  | Index | Unit                    | Description |  |
|----------------|------------|---|-------|-------------------------|-------------|--|
| A4N compatible | Stand ar d |   |       |                         |             |  |
|                |            | (expansion portion)   |       |                         |             |  |
| -              | 24h        | Logical output signal (expansion portion)                           | -     | 0                       | -           | Logic level of output signal (expansion portion)   |
| -              | 25h        | Physical input signal   | -     | 0                       | -           | Physical level of input signal   |
| -              | 26h        | Physical output signal  | -     | 0                       | -           | Physical level of output signal  |
| 131h           | 31h        | Inertia ratio   | -     | 0                       | %           | The ratio of load inertia to the motor's rotor inertia (equivalent of value in Pr 0.04) Inertia ratio<br>= (load inertia / rotor inertia) × 100  |
| 132h           | 32h        | Automatic motor recognition   | -     | 0                       | -           | 0: Invalid<br>1: Valid   |
| 133h           | 33h        | Cause of no revolution  | -     | 0                       | -           | The number which shows the cause that the motor is not running.  |
| 134h           | 34h        | Warning flags   | -     | 0                       | -           | The flag that shows the state of the warning currently occurring.<br>* The corresponding bit is set to 1 to activate the flag (showing warning status).  |
| -              | 37h        | Multiple alarm occurrences /Warning information <sup>(Note 1)</sup> | -     | Refer to Section 6-9-6. | -           | Information of all the alarms or warnings currently occurring  |
| 201h           | 41h        | Mechanical angle (Single turn data)                                 | -     | 0                       | pulse       | The mechanical angle (one revolution data of an absolute encoder) of the motor<br>* The polarity is fixed and data increases at CCW rotation.<br><div style="border: 1px solid black; padding: 2px; width: fit-content;">One revolution data = 0 to (Encoder resolution - 1)</div> |
| 202h           | 42h        | Electrical angle  | -     | 0                       | 0.7031°     | Motor electrical angle<br>* The polarity is fixed and data increases at CCW rotation.<br><div style="border: 1px solid black; padding: 2px; width: fit-content;">Electrical angle = 0 to 1FF [Hex]</div>   |
| -              | 43h        | Multi-turn data   | -     | 0                       | Turn        | Multi-turn data of the absolute encoder<br>* In the incremental mode (Pr 0.15 = 1), multi-turn data becomes an indefinite value.   |
| -              | 44h        | Encoder status <sup>(Note 1)</sup>                                  | -     | 0                       | -           | The status of the encoder  |
| -              | 47h        | Encoder pulse   | -     | 0                       | pulse       | The sum of encoder feedback pulses   |



## 12.4 Monitor Commands

| Type_Code      |            | Name   | Index | Unit | Description            |  |
|----------------|------------|--|-------|------|------------------------|--|
| A4N compatible | Stand ar d |  |       |      |                        |  |
|                |            | sum <sup>(Note 1)</sup>                              |       |      |                        |  |
| -              | 48h        | External scale pulse sum <sup>(Note 1)</sup>         | -     | 0    | Pulse (External scale) | The sum of external scale feedback pulses  |
| -              | 49h        | External scale absolute position <sup>(Note 1)</sup> | -     | 0    | Pulse (External scale) | The absolute position of the external scale  |
| -              | 61h        | Power on cumulative time                             | -     | -    | 30 min                 | Cumulative on-time of control power to the servo amplifier<br>* Because the power ON time is recored in unit of 30 minutes, a turn-on period shorter than 30 minutes is not recorded in the cumulative on-time.<br>not recorded in the cumulative on-time. |

(Note 1) The version before the function extended version 1 is not supported.

(Note 2) Be careful that the unit is different from that used for A4N and A5N. (A4N, A5N: [0.1%], A6N: [%])

\* With the function extended version 3 or higher, the unit can be changed through bit 7 of Pr 7.99.

Pr7.99 bit7 0: [%], 1: [0.1%]

| Type_Code      |            | Name   | Index | Unit | Description |   |
|----------------|------------|--|-------|------|-------------|---|
| A4N compatible | Stand ar d |  |       |      |             |   |
| -              | 62h        | Servo amplifier temperature                  | -     | -    | °C          | Temperature inside the servo amplifier  |
| -              | 63h        | Encoder temperature                          | -     | -    | °C          | Temperature inside the encoder<br>* Applicable only to 23-bit encoder. 0 for unsupported encoder.   |
| -              | 64h        | Number of inrush resistance relay operations | -     | -    | Cycle       | Operating cycles of inrush current suppression resistor relay<br>* Saturation will occur at maximum value of 40000000h.<br>* Because the power ON time is recored in unit of 30 minutes, a turn-on period shorter than 30 minutes is not recorded in the cumulative cycles. |
| -              | 65h        | No. of dynamic brake operations              | -     | -    | Cycle       | Number of operations of dynamic brake relay<br>* Saturation will occur at maximum value of 40000000h.<br>* Because the power ON time is recored in unit of 30 minutes, a turn-on period shorter than 30 minutes is not recorded in the cumulative time.                     |
| -              | 66h        | Fan operating time                           | -     | -    | 30 min      | Operating time of cooling fan<br>* Because the power ON time is recored in unit of 30 minutes, a turn-on period   |

## 12.4 Monitor Commands

| Type_Code      |           | Name  | Index | Unit | Description  |  |
|----------------|-----------|---|-------|------|--|--|
| A4N compatible | Stand ard |   |       |      |  |  |
|                |           |   |       |      | shorter than 30 minutes is not recorded in the cumulative time.<br>* 0 when no fan is installed. |  |
| -              | 67h       | Fan life expectancy   | -     | -    | 0.1%   | Percent of fan life expectancy<br>* Because the power ON time is recorded in unit of 30 minutes, a turn-on period shorter than 30 minutes is not recorded in the cumulative time.<br>* 0 when no fan is installed.                     |
| -              | 68h       | Capacitor life expectancy   | -     | -    | 0.1%   | Percent of life expectancy of main power source capacitor<br>* Because the power ON time is recorded in unit of 30 minutes, a turn-on period shorter than 30 minutes is not recorded in the cumulative time.                           |
| -              | 69h       | Voltage across PN   | -     | -    | V  | Main power source PN voltage   |
| -              | 6Ch       | Consumed power of motor<br>(Note 1)   | -     | -    | W  | Momentary power consumption of the motor   |
| -              | 6Dh       | Motor power consumption<br>(Note 1)   | -     | -    | Wh   | Power consumption of the motor   |
| -              | 6Eh       | Cumulative motor power consumption<br>(Note 1)                              | -     | -    | Wh   | Cumulative value of motor power consumption  |
| 401h           | 71h       | RTEX Cumulative communication errors  | -     | 0    | Cycle  | Cumulative number of RTEX communication errors<br>* Saturation will occur at maximum value of FFFFh.<br>The count will be cleared upon restarting of servo amplifier or resetting of control power source.                             |
| -              | 77h       | RTEX UpdateCounter cumulative error count<br>(Note 1)                       | -     | 0    | Cycle  | Cumulative number of communication errors of RTEX UpdateCounter<br>* Saturation will occur at maximum value of 7FFFh.<br>The count will be cleared upon restarting of servo amplifier or resetting of control power source.            |
| -              | 78h       | RTEX communication Cumulative RTEX communication timeout errors<br>(Note 1) | -     | 0    | Cycle  | Cumulative number of RTEX communication data reception interruption errors<br>* Saturation will occur at maximum value of FFFFh.<br>The count will be cleared upon restarting of servo amplifier or resetting of control power source. |

| Type_Code      |           | Name                                    | Index | Unit | Description |  |
|----------------|-----------|---|-------|------|-------------|--|
| A4N compatible | Stand ard |   |       |      |             |  |
| 411h           | 81h       | Encoder cumulative communication errors | -     | 0    | Cycle       | Cumulative number of communication errors between encoders<br>* Saturation will occur at maximum value of FFFFh.<br>The count will be cleared upon restarting of servo amplifier or resetting of control power source. |

(Note 1) The version before the function extended version 1 is not supported.

| Type_Code      |           | Name  | Index | Unit | Description               |  |
|----------------|-----------|---|-------|------|---------------------------|--|
| A4N compatible | Stand ard |   |       |      |                           |  |
| 413h           | 83h       | External scale cumulative communication errors <sup>(Note 1)</sup>    | -     | 0    | Cycle                     | Cumulative number of communication errors between external scales<br>* Saturation will occur at maximum value of FFFFh.<br>The count will be cleared upon restarting of servo amplifier or resetting of control power source.  |
| -              | 84h       | External scale abnormal communication data errors <sup>(Note 1)</sup> | -     | 0    | Cycle                     | Cumulative number of communication data errors in communication between external scales<br>* Saturation will occur at maximum value of FFFFh.<br>The count will be cleared upon restarting of servo amplifier or resetting of control power source.  |
| -              | 85h       | For manufacturer's use  | -     | -    | -                         | -  |
| -              | 86h       | Hybrid position deviation <sup>(Note 1)</sup>                         | -     | -    | Command unit              | Tolerance between encoder position and external scale position   |
| -              | 87h       | External scale data <sup>(Note 1)</sup><br>(Leftmost 24 bits)         | -     | 0    | Pulse<br>(External scale) | Rightmost 24 bits of external scale data   |
| -              | 88h       | External scale data <sup>(Note 1)</sup><br>(Rightmost 24 bits)        | -     | 0    | Pulse<br>(External scale) | <Virtual full-close control mode function disabled><br>Leftmost 24 bits of external scale data is output.<br><Virtual full-close control mode function enabled><br><ul style="list-style-type: none"> <li>When an AB-phase output type scale is connected, position data (16 bits) is output that is set to 0 when the power is turned ON. Note that it is not affected by Pr 3.26 Reversal of direction.</li> </ul> |

## 12.4 Monitor Commands

| Type_Code      |            | Name  |   | Index | Unit                        | Description  |
|----------------|------------|---|---|-------|-----------------------------|--|
| A4N compatible | Stand ar d |   |   |       |                             |  |
|                |            |   |   |       |                             | <ul style="list-style-type: none"> <li>When a serial incremental scale is connected, position data (24 bits) of the serial incremental scale is output. Note that the data output is position data affected by Pr 3.26 Reversal of direction.</li> </ul> |
| -              | 89h        | External scale status <sup>(Note 1)</sup>                                 | - | 0     | -                           | Status of external scale   |
| -              | A1h        | Velocity control command <sup>(Note 1)</sup>                              | - | 0     | r/min                       | Velocity control command   |
| -              | A5h        | Internal position command speed <sup>(Note 1)</sup>                       | - | 0     | r/min                       | Internal position command speed  |
| -              | A6h        | Speed deviation <sup>(Note 3)</sup>                                       | - | 0     | r/min                       | Speed deviation  |
| -              | A8h        | Positive direction torque limit value <sup>(Note 1)</sup>                 | - | 0     | 0.05%                       | Positive direction torque limit value  |
| -              | A9h        | Negative direction torque limit <sup>(Note 1)</sup>                       | - | 0     | 0.05%                       | Negative direction torque limit value  |
| -              | AAh        | Speed limit value <sup>(Note 1)</sup>                                     | - | 0     | r/min                       | Speed limit value  |
| -              | ABh        | Gain switching flag <sup>(Note 1)</sup>                                   | - | 0     | -                           | Gain switching flag  |
| -              | B1h        | Deterioration diagnosis state <sup>(Note 1)</sup>                         | - | 0     | -                           | Deterioration diagnosis state  |
| -              | B2h        | Deterioration diagnosis torque average time <sup>(Note 1)</sup>           | - | 0     | 0.1%<br><sup>(Note 2)</sup> | Deterioration diagnosis torque command average time  |
| -              | B3h        | Deterioration diagnosis torque command standard value <sup>(Note 3)</sup> | - | 0     | 0.1%                        | Deterioration diagnosis torque command standard value  |
| -              | B4h        | Deterioration diagnosis inertia ratio estimate <sup>(Note 1)</sup>        | - | 0     | %                           | Deterioration diagnosis inertia ratio estimate   |

(Note 1) The version before the function extended version 1 is not supported.

(Note 2) Be careful that the unit is different from the one of the data displayed on the setup support software (PANATERM).

## 12.4 Monitor Commands

(Note 3) The version before the function extended version 2 is not supported.

| Type_Code      |          | Name  | Index | Unit | Description              |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
|----------------|----------|---|-------|------|--------------------------|---|------|-----|-------------|--------|--------|------------------------|--------|--------|------------------------|--------|--------|------------------------|---|---|---|---|--------|------------------------|--------|--------|------------------------|
| A4N compatible | Standard |   |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
| -              | B5h      | Deterioration diagnosis unbalanced load estimate<br>(Note 1)                      | -     | 0    | 0.1% *2)                 | Deterioration diagnosis unbalanced load estimate  |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
| -              | B6h      | Deterioration diagnosis unbalanced load estimate<br>(Note 1)                      | -     | 0    | 0.1% *2)                 | Deterioration diagnosis unbalanced load estimate  |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
| -              | B7h      | Deterioration diagnosis unbalanced load estimate<br>(Note 1)                      | -     | 0    | 0.1% / (10000 r/min) *2) | Deterioration diagnosis unbalanced load estimate  |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
| -              | FAh      | Monitor flag (Note 1)   | -     | 0    | -                        | Various flag information of the servo amplifier<br>The contents of Monitor_Data, the response data, are as follows. <table border="1" data-bbox="847 973 1249 1591"> <thead> <tr> <th>Byte</th> <th>bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>12, 20</td> <td>7 to 0</td> <td>For manufacturer's use</td> </tr> <tr> <td>13, 21</td> <td>7 to 0</td> <td>For manufacturer's use</td> </tr> <tr> <td rowspan="4">14, 22</td> <td>7 to 6</td> <td>For manufacturer's use</td> </tr> <tr> <td>5</td> <td>Semi-closed / full-closed selection state<br/>0: Semi-closed<br/>1: Full-closed</td> </tr> <tr> <td>4</td> <td>Incremental / absolute selection state<br/>0: Incremental mode<br/>1: Absolute mode</td> </tr> <tr> <td>3 to 0</td> <td>For manufacturer's use</td> </tr> <tr> <td>15, 23</td> <td>7 to 0</td> <td>For manufacturer's use</td> </tr> </tbody> </table> | Byte | bit | Description | 12, 20 | 7 to 0 | For manufacturer's use | 13, 21 | 7 to 0 | For manufacturer's use | 14, 22 | 7 to 6 | For manufacturer's use | 5 | Semi-closed / full-closed selection state<br>0: Semi-closed<br>1: Full-closed | 4 | Incremental / absolute selection state<br>0: Incremental mode<br>1: Absolute mode | 3 to 0 | For manufacturer's use | 15, 23 | 7 to 0 | For manufacturer's use |
| Byte           | bit      | Description   |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
| 12, 20         | 7 to 0   | For manufacturer's use  |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
| 13, 21         | 7 to 0   | For manufacturer's use  |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
| 14, 22         | 7 to 6   | For manufacturer's use  |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
|                | 5        | Semi-closed / full-closed selection state<br>0: Semi-closed<br>1: Full-closed     |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
|                | 4        | Incremental / absolute selection state<br>0: Incremental mode<br>1: Absolute mode |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
|                | 3 to 0   | For manufacturer's use  |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |
| 15, 23         | 7 to 0   | For manufacturer's use  |       |      |                          |   |      |     |             |        |        |                        |        |        |                        |        |        |                        |   |   |   |   |        |                        |        |        |                        |

(Note 1) The version before the function extended version 2 is not supported.

(MEMO)

## Revision History

The manual code is shown at the bottom of the cover page.

| Date of issue | Manual No.     | Revision details   |
|---------------|----------------|--|
| February 2021 | WUME-GM1PGR-01 | 1st Edition  |
| August 2021   | WUME-GM1PGR-02 | 2nd Edition<br>Added instructions related to EtherCAT.<br>Changed PMC_ReadLatchPosition parameters.<br>Added instructions related to PID control.<br>Added instructions related to the GM1 Pulse Output Unit.  |
| March 2022    | WUME-GM1PGR-03 | 3rd Edition<br>Clerical corrections<br>Added function block argument for pulse output unit <ul style="list-style-type: none"><li>• Added switching of P-point / E-point control for positioning control</li><li>• Added the creep speed of home return</li></ul> |
| April 2022    | WUME-GM1PGR-04 | 4th edition <ul style="list-style-type: none"><li>• Changed the Company name</li></ul>   |

(MEMO)



(MEMO)

---

Please contact .....

**Industrial Device Business Division,  
Panasonic Industry Co., Ltd.**

7-1-1 Morofuku, Daito City, Osaka, 574-0044, Japan  
[industrial.panasonic.com/ac/e/](http://industrial.panasonic.com/ac/e/)

© Panasonic Industry Co., Ltd 2021-2022

WUME-GM1PGR-04