

ORIENTAL MOTOR Co., Ltd.  
Stepping Motor and Driver Package αSTEP  
High-efficiency AR Series FLEX  
Built-in Controller Type (AC/DC power input)

Sample Screen Manual

Mitsubishi Electric Corporation

## Using the Samples

---

The sample screen data and files such as the instruction manual can be used upon agreement to the following matters.

- (1) This data is available for use by customers currently using or considering use of Mitsubishi products.
- (2) The intellectual property rights of the files provided by Mitsubishi (hereinafter referred to as the "Files") belong to Mitsubishi.
- (3) Alteration, reproduction, transfer or sales of the Files is prohibited.  
This does not apply when the content, in part or full, is used for Mitsubishi products incorporated in a device or system created by the customer. Furthermore, this does not apply to the transfer, reproduction, reference or change of layout in the specifications, designs or instruction manuals of built-in products prepared by the customer using Mitsubishi products.
- (4) Mitsubishi will not be held liable for any damages resulting from the use of the Files or the data extracted from the Files. The customer is responsible for all use.
- (5) If any usage conditions are appended to the Files, those conditions must be observed.
- (6) The Files may be deleted or the contents changed without prior notice.
- (7) When using the Files, please always read the corresponding manuals and related manuals indicated therein. Please pay special attention to safety, and correctly handle the product.

## CONTENTS

---

REVISIONS .....	5
1. OUTLINE .....	6
2. SYSTEM CONFIGURATION .....	6
3. GOT .....	7
3.1 System Applications That Are Automatically Selected .....	7
3.2 Controller Setting of Screen Design Software .....	7
3.3 Overlap Window Setting of Screen Design Software .....	7
4. DRIVER .....	8
4.1 Communication Settings for the AC power input driver .....	8
4.2 Communication Settings for the DC power input driver .....	8
4.3 Supported driver for the sample .....	8
5. SCREEN SPECIFICATIONS .....	9
5.1 Display Language.....	9
5.2 Screen List/Transition.....	9
5.3 Explanation of Screens.....	15
5.3.1 AC/DC Selection Screen (B-30001).....	15
5.3.2 Menu (B-30002).....	16
5.3.3 Operation data (B-31002) .....	17
5.3.4 Parameter Menu 1 (B-30003) .....	19
5.3.5 Parameter I/O (B-31004).....	20
5.3.6 Parameter Motor (B-31005).....	21
5.3.7 Parameter Operation (B-31006).....	22
5.3.8 Parameter Return-to-home (B-31007).....	23
5.3.9 Parameter Alarm & Warning (B-31008).....	24
5.3.10 Parameter Menu 2 (B-30004) .....	25
5.3.11 Parameter Coordinate (B-31010) .....	26
5.3.12 Parameter Common (B-31011) .....	27
5.3.13 Parameter I/O Function (B-31012) .....	28
5.3.14 Parameter I/O Function RS-485(B-31013) .....	29
5.3.15 Parameter Communication (B-31014).....	30
5.3.16 Monitor Menu (B-30005) .....	31
5.3.17 Monitor Status (B-31015) .....	32
5.3.18 Monitor I/O Monitor (B-31016).....	33
5.3.19 Monitor I/O RS-485 (B-31017).....	34
5.3.20 Monitor Alarm Records (B-31018).....	35
5.3.21 Monitor Warning Records (B-31019).....	36
5.3.22 Manual Display (B-30500) .....	37
5.3.23 Test Menu (B-30006) .....	39
5.3.24 Test I/O Test (B-31020).....	40
5.3.25 Test Operation (B-31021).....	41
5.3.26 Test Multiple Operation (B-31022 to 31029) .....	43
5.3.27 System Data management (B-31030) .....	45
5.3.28 Alarm Reset (W-30001) .....	46
5.3.29 Language Setting (W-30002) .....	47
5.3.30 Clock Setting (W-30003).....	48
5.3.31 Axis Switching (W-30004).....	49
5.3.32 Operation Data Input (W-32001).....	50
5.3.33 STOP Input Action (W-32002) .....	51
5.3.34 Home-seeking Mode (W-32003) .....	52
5.3.35 IN Input Function Selection (W-32004) .....	53
5.3.36 NET-IN Input Function Selection (W-32005).....	54
5.3.37 OUT/NET-OUT Function Selection (W-32006) .....	55
5.3.38 Alarm Occurring Confirmation (W-32007).....	56
5.3.39 Motor Running Confirmation (W-32008) .....	57

5.4	Device List.....	58
5.5	Comment List .....	64
5.6	Script List.....	66
6.	MANUAL DISPLAY .....	180
6.1	Preparing Document Data for Manual Display .....	180
7.	SAVE/READ OPERATION DATA AND PARAMETER TO/FROM A USB MEMORY OR A CF CARD .....	181
7.1	Ways of Saving Operation Data and Parameters to a USB Memory or a CF Card.....	181
7.2	Ways of Reading Operation Data and Parameters from a USB Memory or a CF Card .....	189
8.	SPECIFICATION OF THE MODBUS ADDRESS OF GOT.....	193
8.1	Address Conversion Method .....	193

## REVISIONS

---

### Sample Screen Manual

Date	Control No.*	Description
2014/8	BCN-P5999-0416	First edition
2015/6	BCN-P5999-0416-2	Device Specification for Document ID

\* The Control No. is noted at the lower right of each page.

### Project Data

Date	Project data	GT Designer3*	Description
2014/8	ORIENTAL_AR-MODBUS_V_Ver1_E.GTX	1.117X	First edition
2015/6	ORIENTAL_AR-MODBUS_V_Ver2_E.GTX	1.128J	Device Specification for Document ID

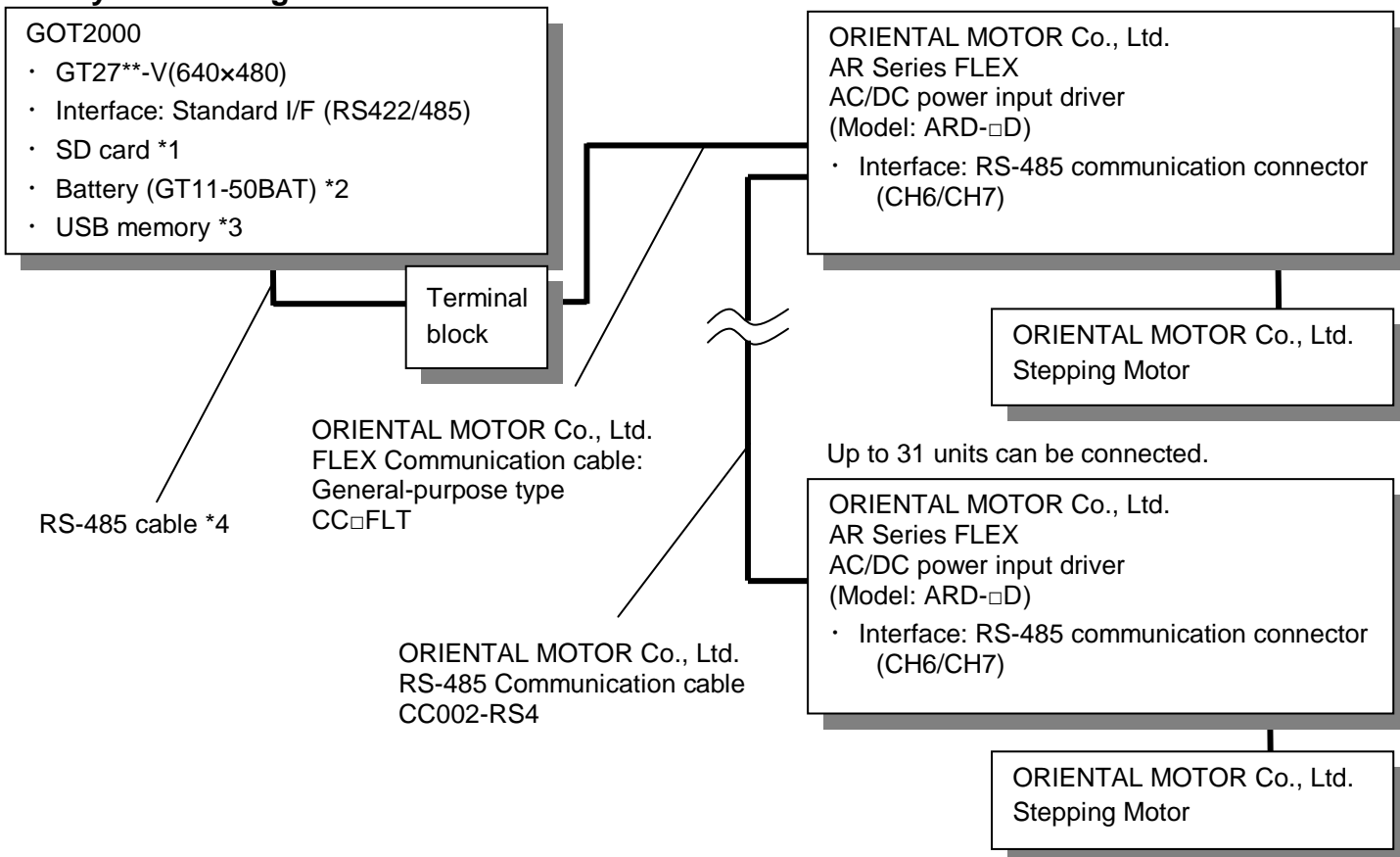
\* The version number of screen design software used to create the project data is listed. Please use the screen design software with the listed version or later.

## 1. OUTLINE

This manual explains the sample screens of GOT2000 connected to an ORIENTAL MOTOR Co., Ltd. AR Series FLEX driver(ARD-□D) in serial (RS-485) communication. The sample screens can be used to monitor or change the current value and setting value.

## 2. SYSTEM CONFIGURATION

### 2.1 System Configuration



\*1: The SD card is used for the document display and recipe functions.

\*2: The battery is used for the backup of the clock data .(The battery is provided with the GOT as standard.)

\*3: The USB memory is used for recipe functions.

\*4: For more details about the connection method, please refer to "GOT2000 Series Connection Manual (Microcomputer, MODBUS, Products, Peripherals)".

### 3. GOT

#### 3.1 System Applications That Are Automatically Selected

Type	System application name		
Standard Function	Standard System Application		
	Standard Font	Japanese	
Communication r	MODBUS/RTU		
Extended Function	Standard Font		Chinese (Simplified)
	Outline Font	Gothic	Alphanumeric/Kana
			Japanese (Kanji)
			Chinese (Simplified)
	Recipe Operation		
	Document Display		

#### 3.2 Controller Setting of Screen Design Software

Item	Set value	Remarks
Transmission Speed(BPS)	115200	
Data Bit	8 bit	
Stop Bit	1 bit	
Parity	Even	
Retry(Times)	2	
Timeout Time(Sec)	3	
Host Address	1	Please set the axis No. of controller to be monitored.
Delay Time(ms)	8	The set value should be greater than or equal to 8ms.
32bit Storage	HL Order	Please set the HL Order.
FunctionCode[0F]	Not used	
FunctionCode[10]	Used	
Coil read times(Points)	2000	
Input relay read times(Points)	2000	
Holding register read times(Points)	16	Up to 16 points
Input register read times(Points)	125	
Coil write times(Points)	800	
Holding register write times(Points)	16	Up to 16 points

#### 3.3 Overlap Window Setting of Screen Design Software

[Close the window when switching base screens] of [Detail Setting] for overlap window in [Screen Switching/Window] is enabled to close the window when switching base screens.

## 4. DRIVER

### 4.1 Communication Settings for the AC power input driver

When our company checks the controller, the set values are as follows.

#### (1) Parameter settings

Item	Set value	Remarks
Communication stop bit	1 bit	
Communication parity	Even number	

#### (2) DIP and rotary switch settings of the driver

Item	Set value	Remarks
Address number setting switch(ID)	ID=1	Change the value for each connected driver.
Transmission rate setting switch(SW2)	115200	SW2=4.
Function setting switch(SW4)	No.2=ON	ON: Choose Modbus protocol
Termination resistor setting switch(TERM.)	No.1,No.2=ON	Only set termination resistor for the driver located farthest away (positioned at the end) to on.

### 4.2 Communication Settings for the DC power input driver

When our company checks the controller, the set values are as follows.

#### (1) Parameter settings

Item	Set value	Remarks
Communication stop bit	1 bit	
Communication parity	Even number	

#### (2) DIP and rotary switch settings of the driver

Item	Set value	Remarks
Address number setting switch(ID)	ID=1	Change the value for each connected driver.
Transmission rate setting switch(SW2)	115200	SW2=4.
Function setting switch(SW4)	No.2=ON	ON: Choose Modbus protocol
Termination resistor setting switch(TERM.)	No.1,No.2=ON	Only set termination resistor for the driver located farthest away (positioned at the end) to on.

### 4.3 Supported driver for the sample

The sample can be used with the driver whose date of production or driver version meet the below conditions.

#### (1) The date of production

January 2014 or after

\*The date of production is recorded on the driver plate.

#### (2) Driver version

AC power input driver : Ver.2.00 and later

DC power input driver : Ver.2.01 and later

\*Driver version can be confirmed with the status and I/O monitor of the data setting software MEXE02.



## 5. SCREEN SPECIFICATIONS

### 5.1 Display Language

The language of the text displayed on the screen can be switched between Japanese, English and Chinese (Simplified). The text strings in each language are registered in the columns No.1 to No.3 in the comment groups No.497 to No.500 as shown below. When the column No. is set in the language switching device, the language corresponding to the column No. will appear.

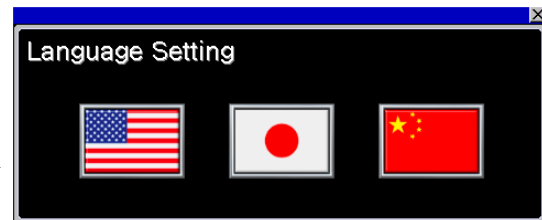
Column No.	Language
1	English
2	Japanese
3	Chinese (Simplified)

### 5.2 Screen List/Transition

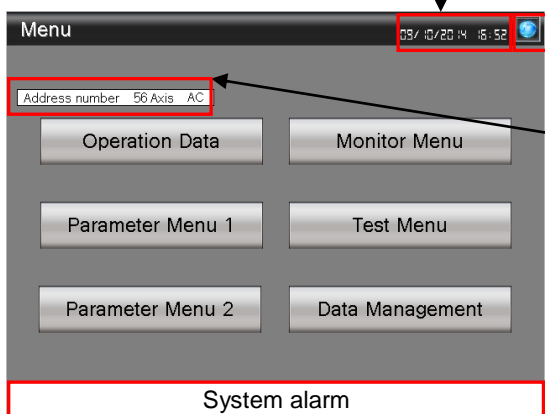
#### 5.2.1 Screen list/transition (common)



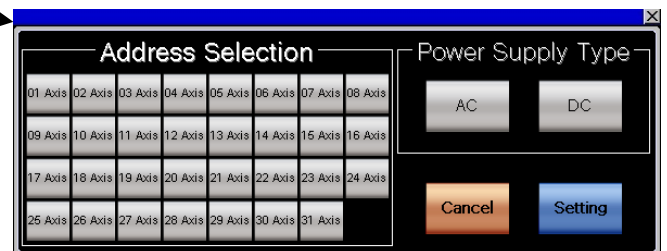
Window screen W-30003 : Clock Setting



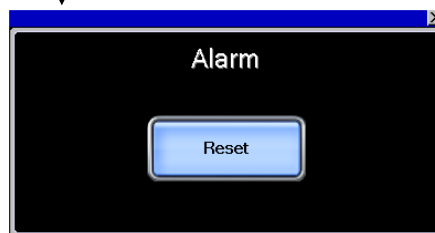
Window screen W-30002 : Language Setting



Base screen : All base screens



Window screen W-30004 : Axis Switching  
(displayed except B-30001)

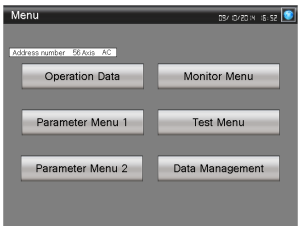


Window screen W-30001 : Alarm Reset

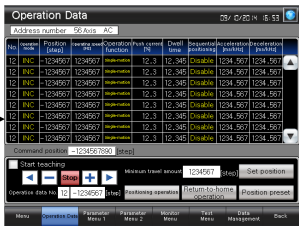
5.2.2 Screen list/transition (individual)



Base screen B-30001 :  
AC/DC Selection Screen



Base screen B-30002 : Menu



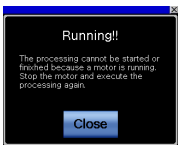
Base screen B-31002 :  
Operation Data



Window screen W-32001 :  
Operation Data Input



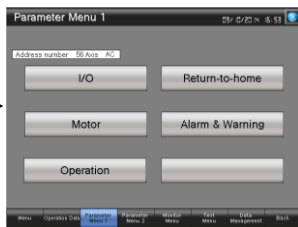
Window screen W-32007 :  
Alarm Occurring Confirmation



Window screen W-32008 :  
Motor Running Confirmation

To next  
page

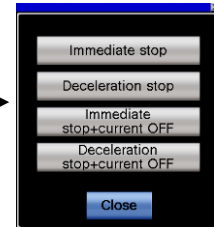
From  
previous  
page



Base screen B-30003 :  
Parameter Menu 1



Base screen B-31004 :  
Parameter I/O



Window screen W-32002 :  
STOP Input Action



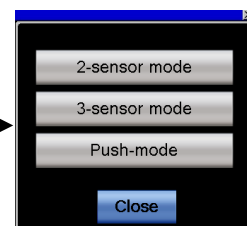
Base screen B-31005 :  
Parameter Motor



Base screen B-31006 :  
Parameter Operation



Base screen B-31007 :  
Parameter Return-to-home



Window screen W-32003 :  
Home-seeking Mode



Base screen B-31008 :  
Parameter Alarm & Warning

To next  
page

From  
previous  
page



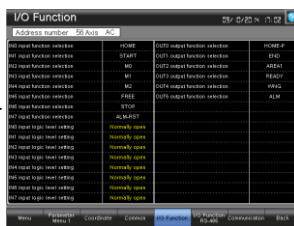
Base screen B-30004 :  
Parameter Menu 2



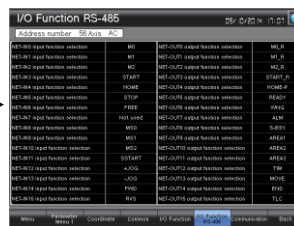
Base screen B-31010 :  
Parameter Coordinate



Base screen B-31011 :  
Parameter Common



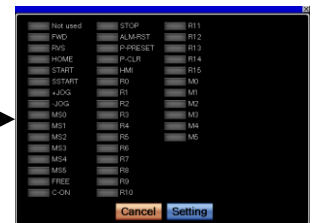
Base screen B-31012 :  
Parameter I/O Function



Base screen B-31013 :  
Parameter I/O Function RS-485



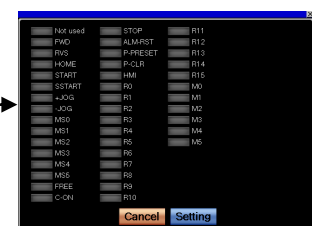
Base screen B-31014 :  
Parameter Communication



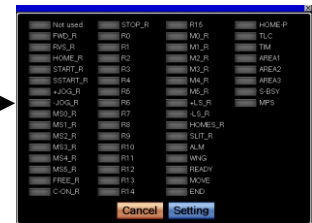
Window screen W-32004 :  
IN Input Function Selection



Window screen W-32006 :  
OUT/NET-OUT  
Output Function Selection



Window screen W-32005 :  
NET-IN Input Function Selection



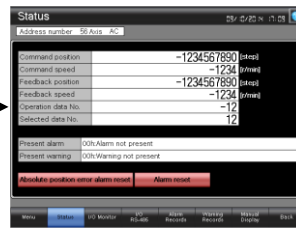
Window screen W-32006 :  
OUT/NET-OUT  
Output Function Selection

To next  
page

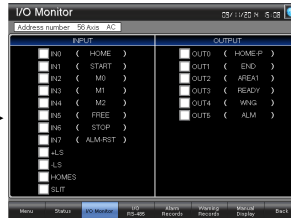
From  
previous  
page



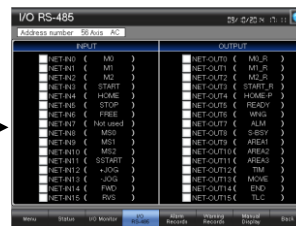
Base screen B-30005 :  
Monitor Menu



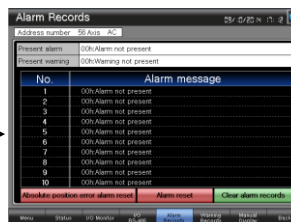
Base screen B-31015 :  
Monitor Status



Base screen B-31016 :  
Monitor I/O Monitor



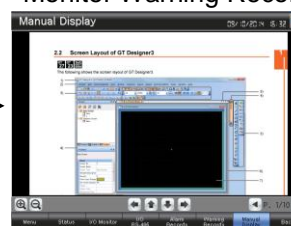
Base screen B-31017 :  
Monitor I/O RS-485



Base screen B-31018 :  
Monitor Alarm Records



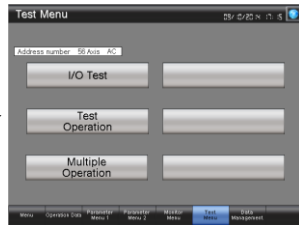
Base screen B-31019 :  
Monitor Warning Records



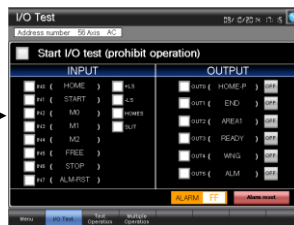
Base screen B-30500 :  
Manual Display

To next  
page

From  
previous  
page



Base screen B-30006 :  
Test Menu



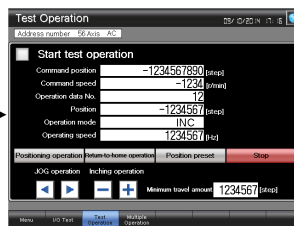
Base screen B-31020 :  
Test I/O Test



Window screen W-32007 :  
Alarm Occurring Confirmation



Window screen W-32008 :  
Motor Running Confirmation



Base screen B-31021 :  
Test Operation



Window screen W-32007 :  
Alarm Occurring Confirmation



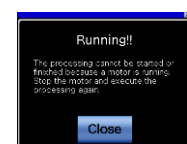
Window screen W-32008 :  
Motor Running Confirmation



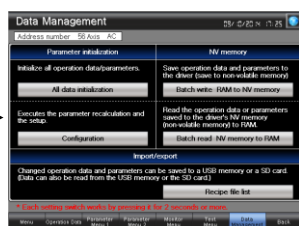
Base screen B-31022 to 31029 :  
Test Multiple Operation



Window screen W-32007 :  
Alarm Occurring Confirmation



Window screen W-32008 :  
Motor Running Confirmation



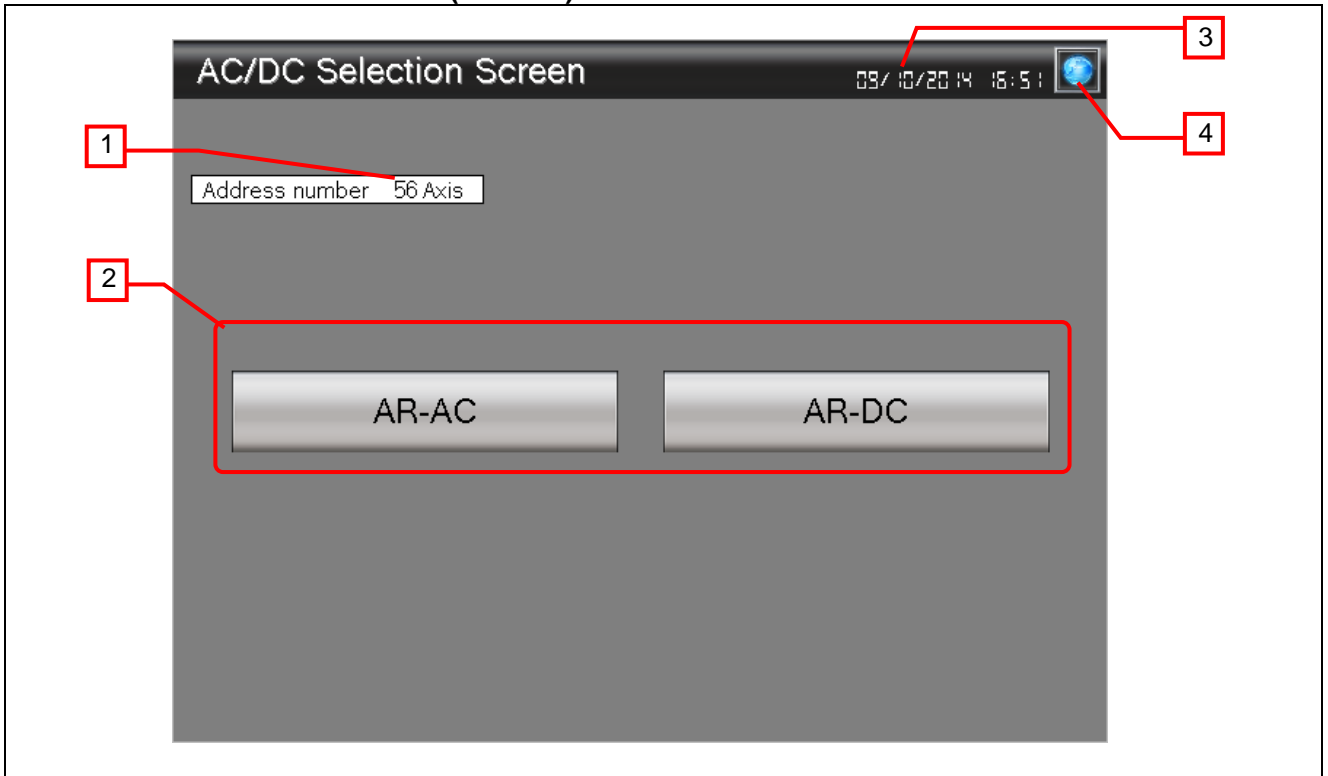
Base screen B-31030 :  
System Data Management



System screen  
Recipe File List Window

## 5.3 Explanation of Screens

### 5.3.1 AC/DC Selection Screen (B-30001)



#### Outline

Select AC/DC power supply.

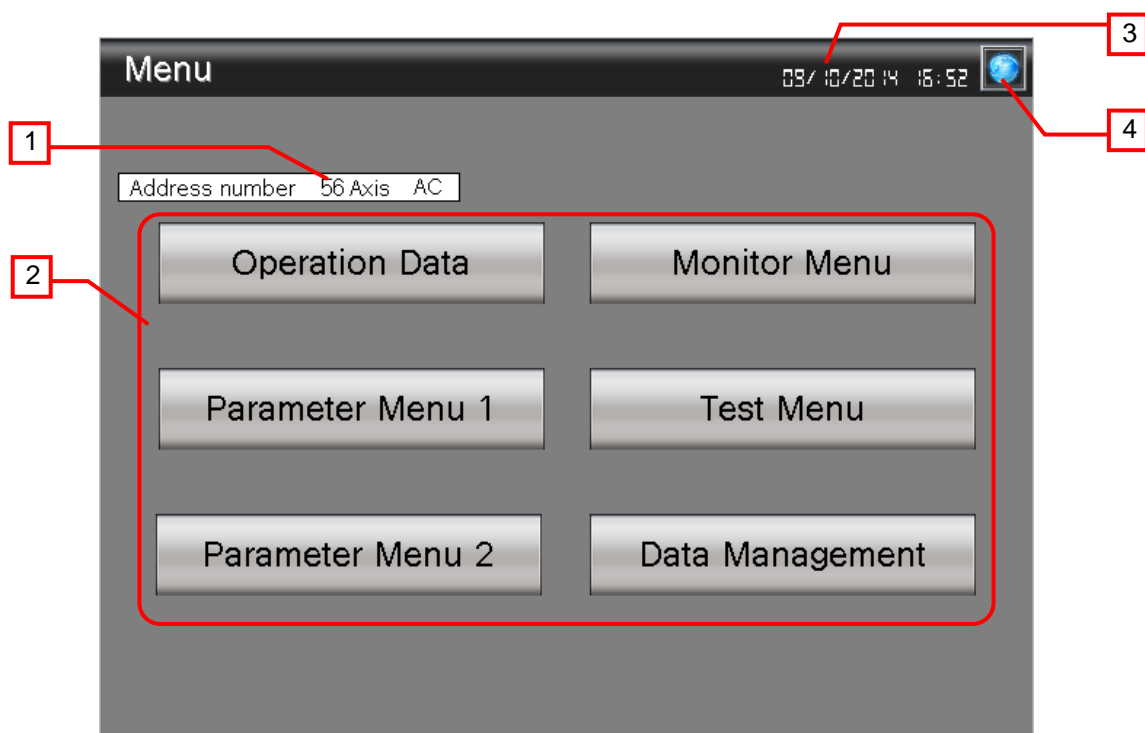
#### Description

1. Displays the address number of the driver to be monitored. The address number can be changed by touching the numerical value.
2. Specifies the power supply type of the driver to be monitored.
3. Displays the current date and time. Touch the button to open the [Clock Setting] window.
4. Opens the [Language Setting] window.

#### Remarks

- When monitoring multiple drivers, the driver with the address number set for [Host Address] of the communication settings must be included. In this sample, "1" is set to the [Host Address]. For more details about the host address setting, please refer to the "GOT2000 Series Connection Manual (Microcomputer, MODBUS Products, Peripherals)".
- When GOT is started, the address number is set to "1" with the project script. For more details about scripts, please refer to "5.6 Script List".
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.2 Menu (B-30002)



#### Outline

This is the Menu screen.

#### Description

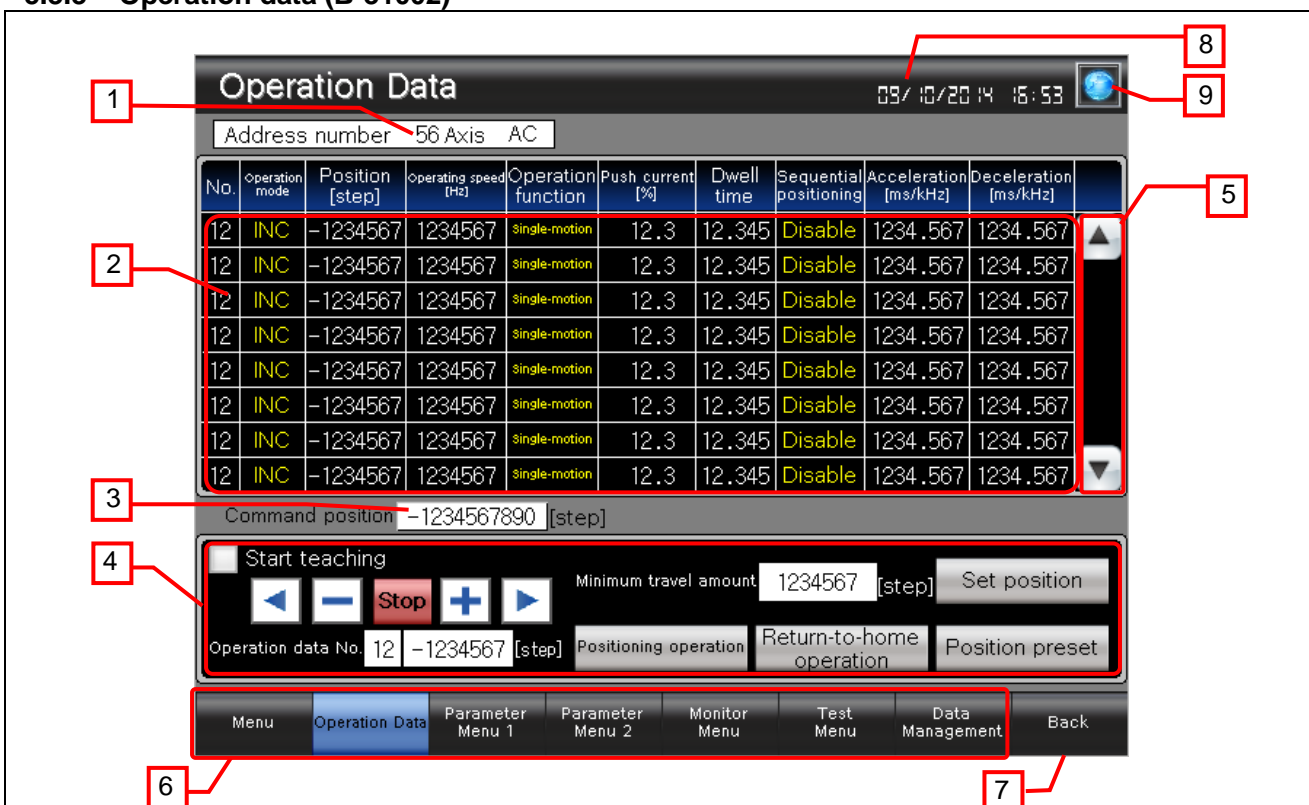
1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Switches to each screen.
3. Displays the current date and time. Touch the button to open the [Clock Setting] window.
4. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.



### 5.3.3 Operation data (B-31002)



#### Outline

This screen displays/changes the operation data of the driver and operates the stepping motor.

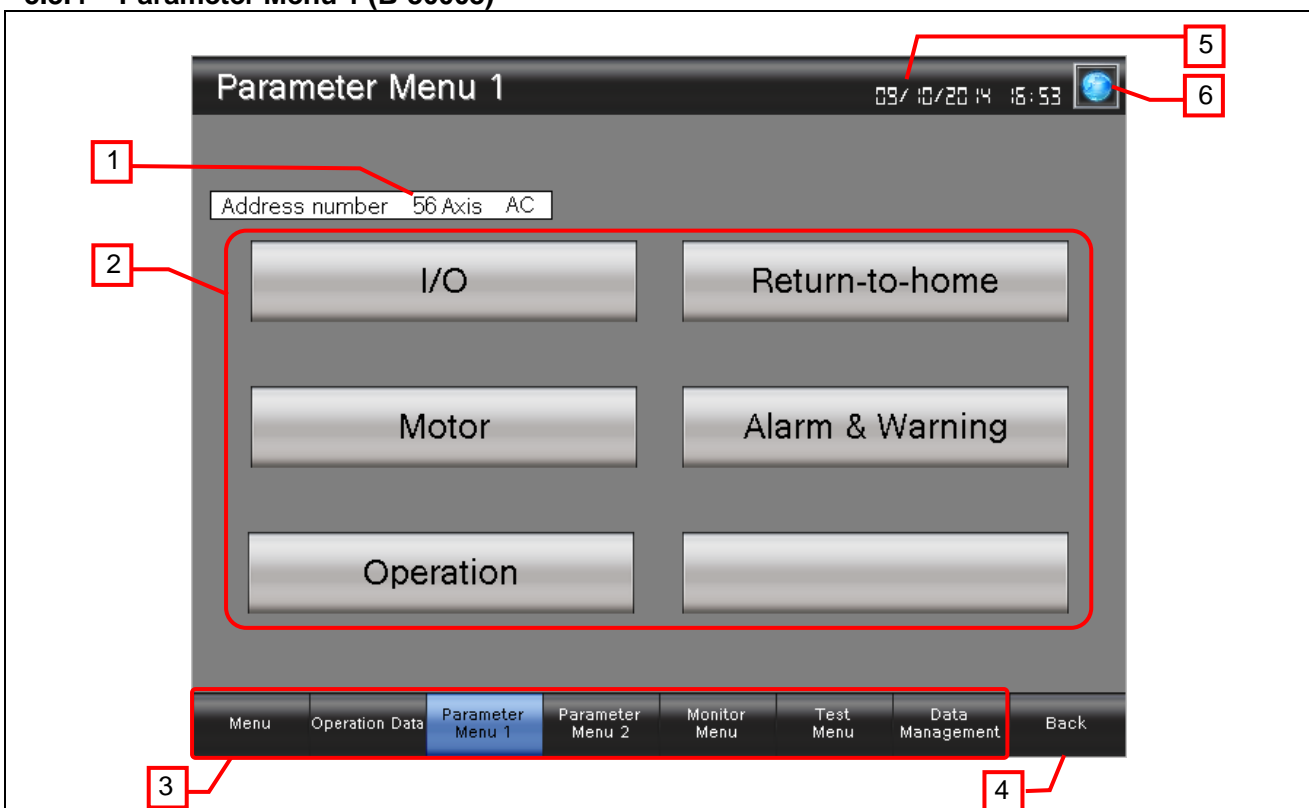
#### Description

- Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
- Displays the operation data. Touch the operation data to open the [Operation data input] window and set related operation data in the window.
- Displays the command position of the motor.
- Check the "Start teaching" to start teaching operation.
  - Minimum travel amount : Sets the minimum travel amount of motor.
  - Operation data No. : Selects the operation data No.
  - : Moves forward/reverse while the button is touched. Operating speed, acceleration, deceleration according to the operation data No.
  - : Adjusts the position of the motor. The travel amount of the motor equals to the minimum travel amount.
  - Stop : Stops the operating motor.
  - Positioning operation : Executes the positioning operation according to the operation data No.
  - Return-to-home operation : Starts the return-to-home operation
  - Set position : Reflects the position of the motor to the position of the operation data No. The operation mode will always be the absolute mode.
  - Position preset : Sets the preset value to the command position. The preset value can be changed in "Preset position" of the [Parameter Coordinate] screen.
- Scrolls the operation data.
  - : Scrolls 8 data up.
  - : Scrolls 8 data down.
- Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
- Switches to the previously opened screen.
- Displays the current date and time. Touch the button to open the [Clock Setting] window.
- Opens the [Language Setting] window.

#### Remarks

- Each time teaching operation starts or completes, momentarily loses the excitation of the motor. Please pay extra attention when using vertical axis and so on. Moreover, the motor excites again after a momentary cutoff, so a position deviation occurs. If position accuracy is required, it is recommended that executing return-to-home operation immediately after teaching operation is completed.
- During teaching operation, acceleration and deceleration can be changed according to the operation data No. as long as acceleration/deceleration type is separate.
- It is not possible to switch screen or change address number when executing teaching operation.
- A screen script is used for executing teaching operation. For more details about scripts, please refer to "5.6 Script List".
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.4 Parameter Menu 1 (B-30003)



#### Outline

This is the Parameter menu 1.

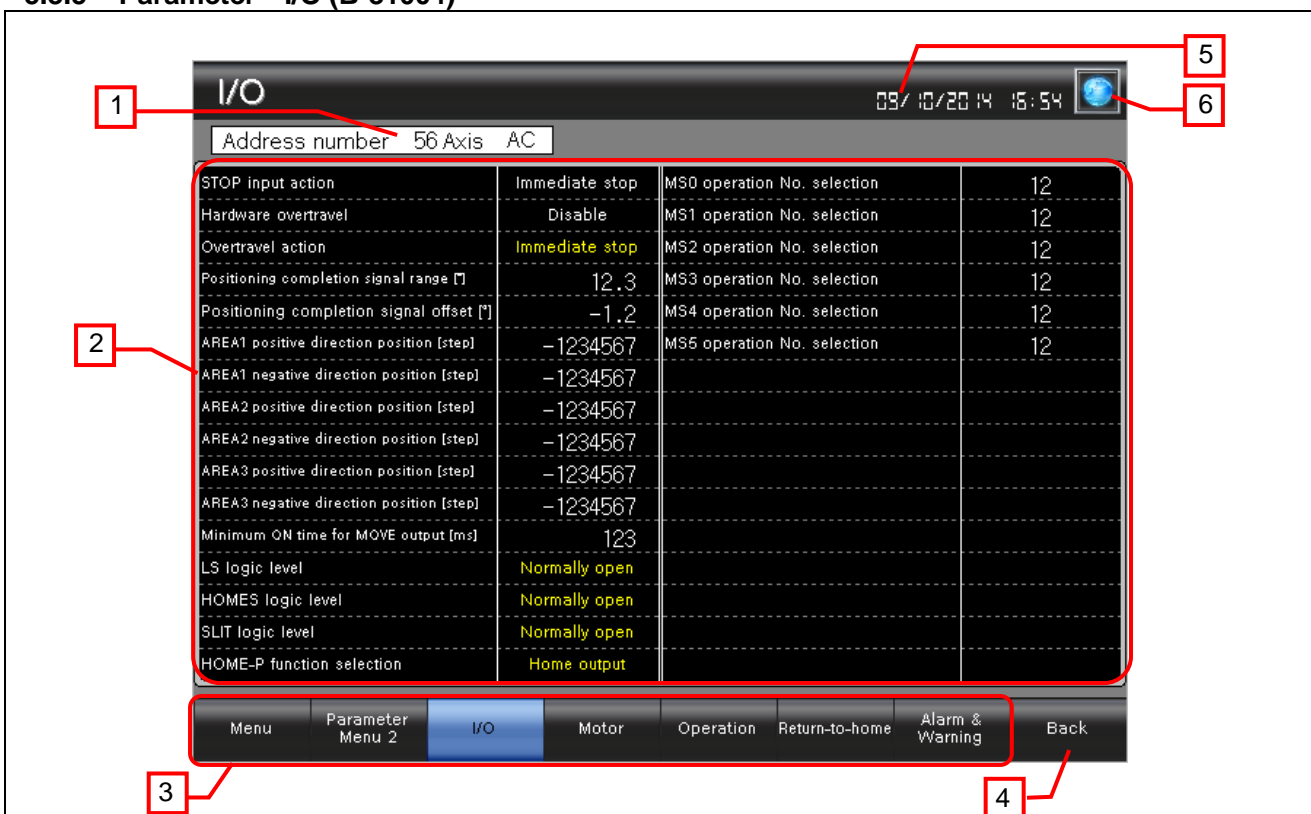
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Switches to each screen.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.5 Parameter I/O (B-31004)



#### Outline

This screen displays and edits the parameters of I/O.

#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the I/O related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If you changed the parameters of LS logic level, HOMES logic level or SLIT logic level, please execute the configuration command.  
The function of the changed signal is not reflected until the configuration command is excuted.  
The configuration command can be executed in the [Data management] screen.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.6 Parameter Motor (B-31005)



#### Outline

This screen displays and edits the parameters of a motor.

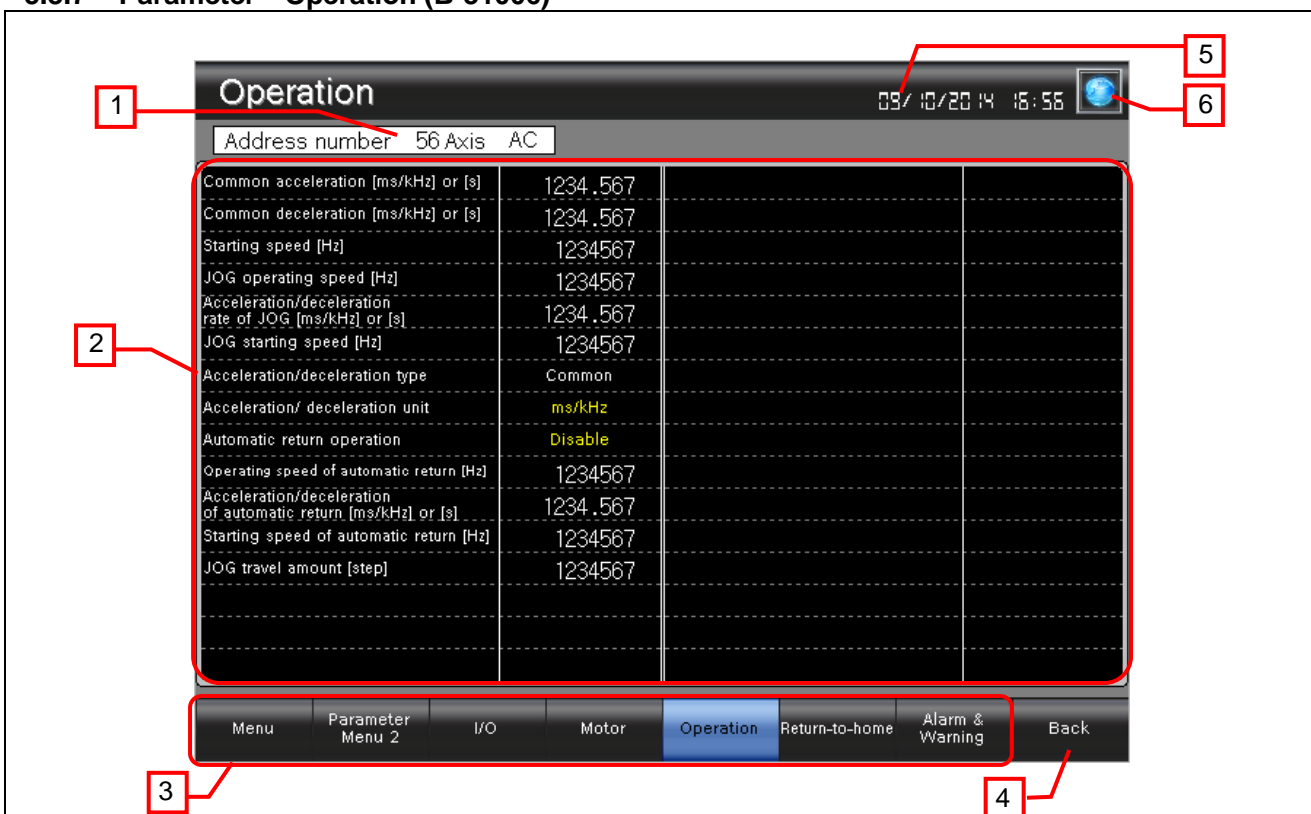
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the motor related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If you changed the parameters of filter selection, control mode or smooth driver, please execute the configuration command.  
The function of the changed signal is not reflected until the configuration command is excuted.  
The configuration command can be executed in the [Data management] screen.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.7 Parameter Operation (B-31006)



#### Outline

This screen displays and edits the parameters of operation.

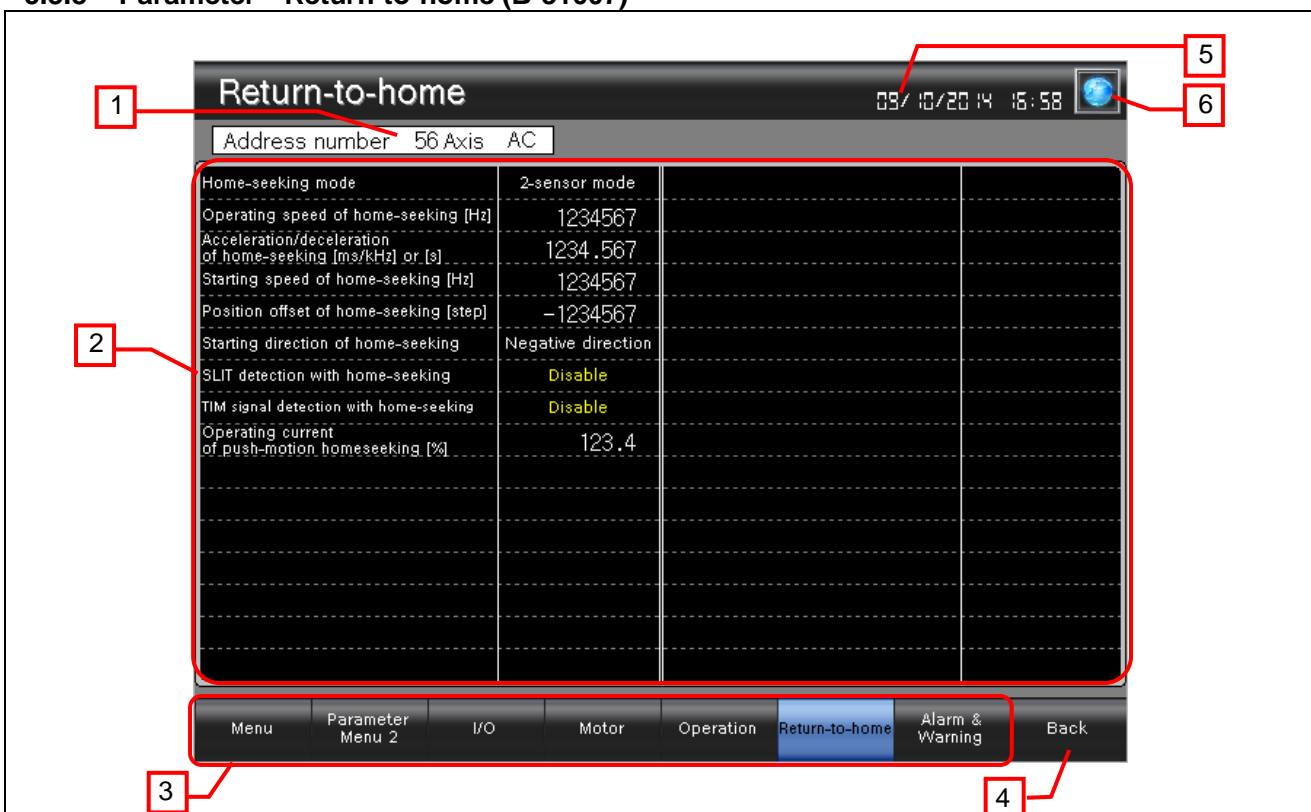
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the operation related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If you changed the parameters of acceleration/deceleration unit or automatic return operation, please execute the configuration command  
The function of the changed signal is not reflected until the configuration command is excuted.  
The configuration command can be executed in the [Data management] screen.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.8 Parameter Return-to-home (B-31007)



#### Outline

This screen displays and edits the parameters of Return-to-home.

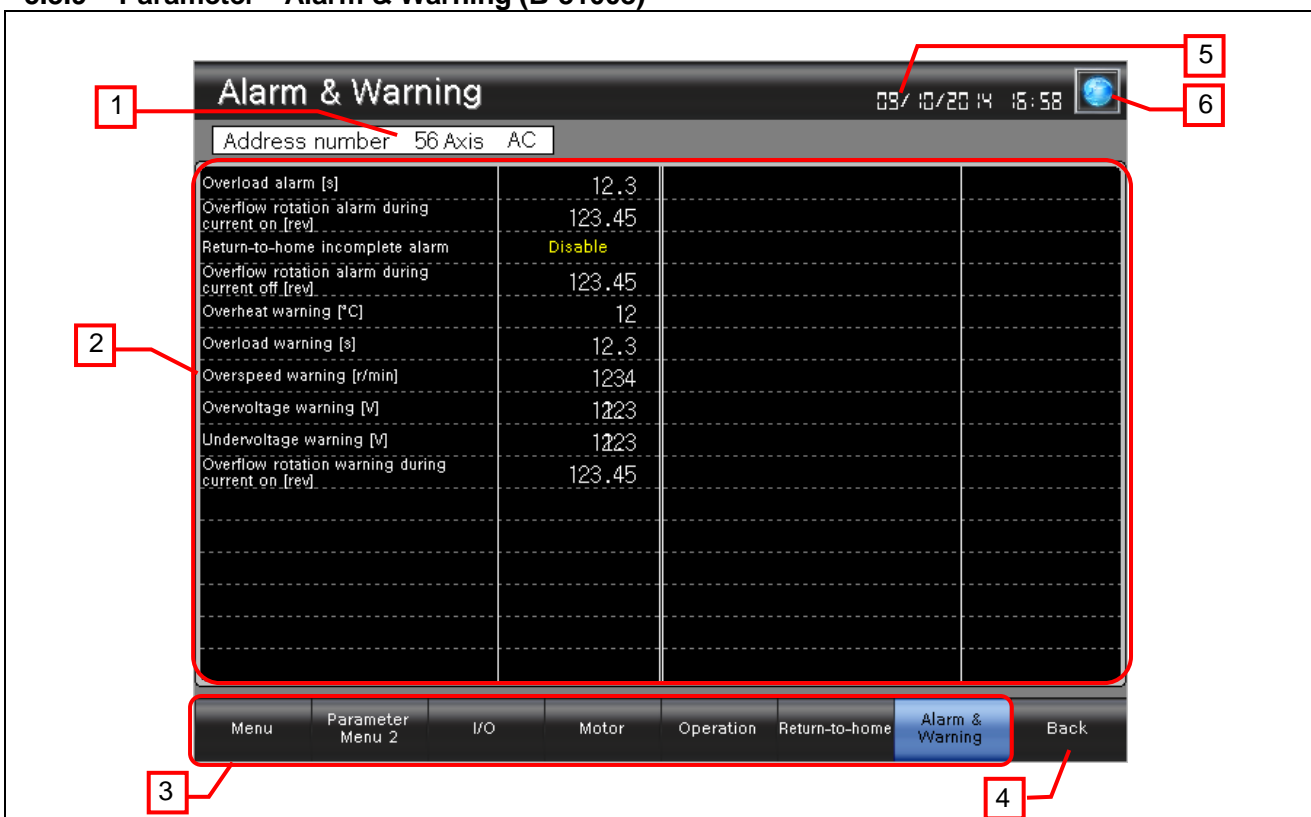
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the return-to-home related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.9 Parameter Alarm & Warning (B-31008)



#### Outline

This screen displays and edits the parameters of alarm & warning.

#### Description

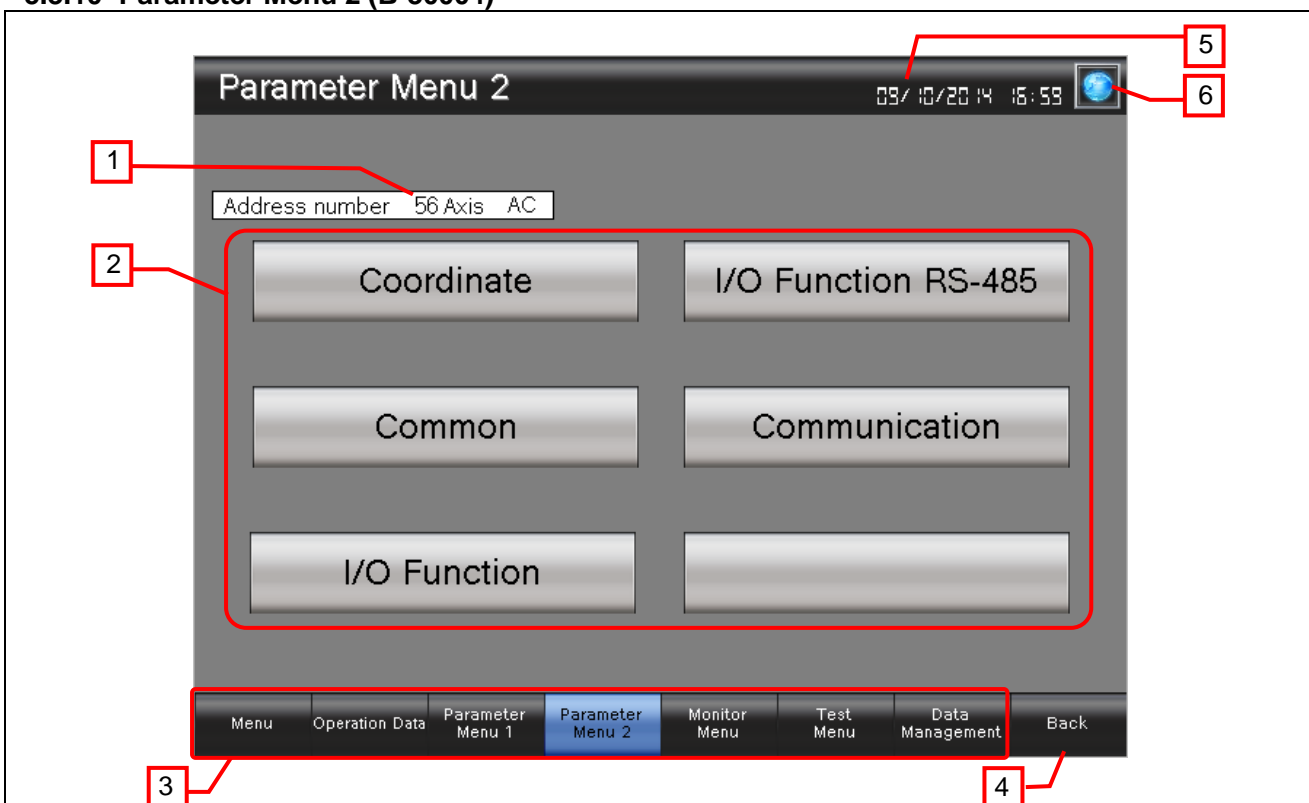
1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the alarm and the warning related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If you changed the parameters of Return-to-home incomplete alarm, please execute the configuration command.  
The function of the changed signal is not reflected until the configuration command is excuted.  
The configuration command can be executed in the [Data management] screen.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.



### 5.3.10 Parameter Menu 2 (B-30004)



#### Outline

This is the parameter menu 2.

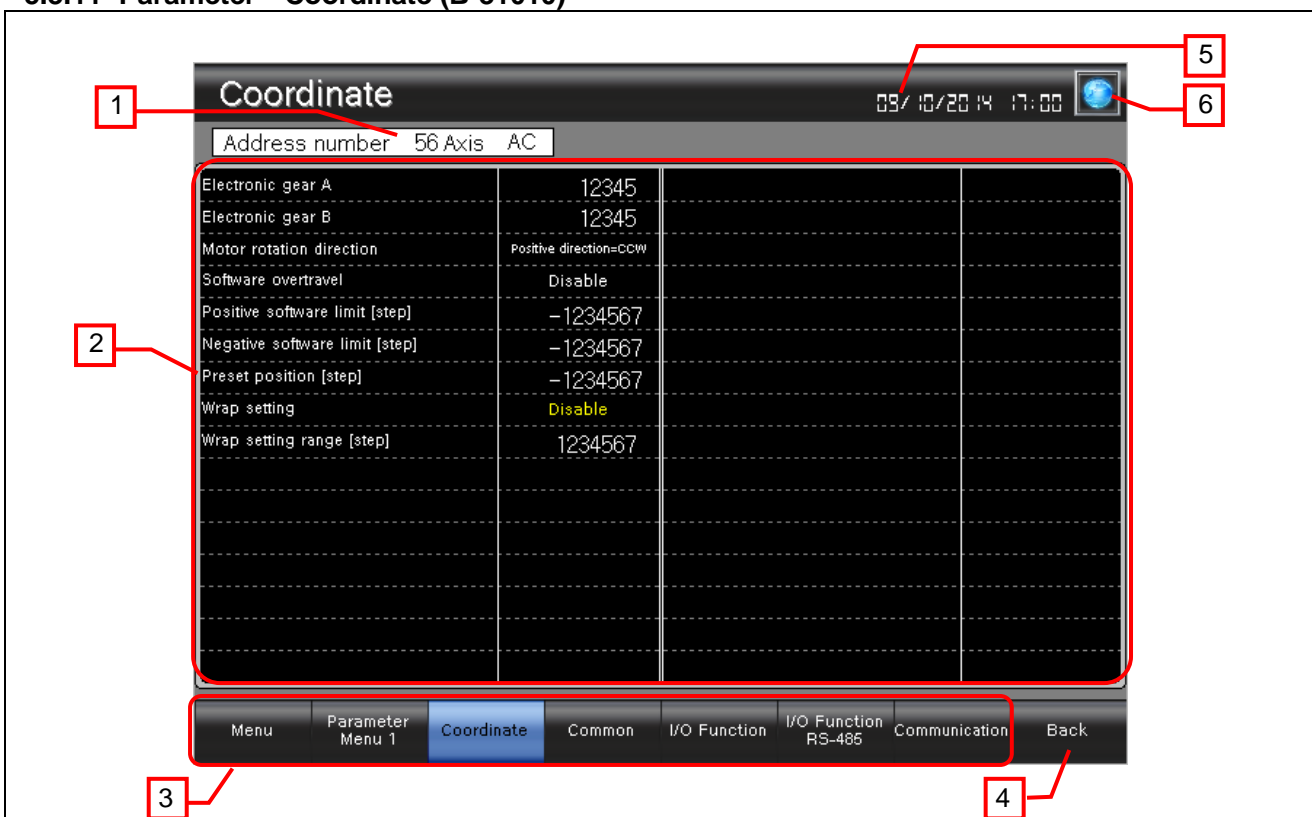
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Switches to each screen.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.11 Parameter Coordinate (B-31010)



#### Outline

This screen displays and edits the parameters of the coordinate.

#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the coordinate related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If you changed the parameters of electronic gear A, electronic gear B, wrap setting, or wrap setting range, please execute the configuration command.  
The function of the changed signal is not reflected until the configuration command is excuted.  
The configuration command can be executed in the [Data management] screen.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

The screenshot displays the 'Common' configuration screen. At the top, the title 'Common' is shown next to a date and time indicator '09/10/2014 17:00'. Below this, there's a section for 'Address number' set to '56 Axis AC'. The central part of the screen features a table with three columns. The first column lists parameters: 'Data setter speed display', 'Data setter edit', and 'Absolute-position backup system'. The second column shows their current settings: 'Signed', 'Disable', and 'Disable' respectively. The third column is currently empty. A red box highlights the entire table area. At the bottom, a navigation bar includes buttons for 'Menu', 'Parameter Menu 1', 'Coordinate', 'Common' (which is highlighted in blue), 'I/O Function', 'I/O Function RS-485', 'Communication', and 'Back'. Red numbered callouts are used to identify key UI components: 1 points to the 'Address number' field; 2 points to the parameter table; 3 points to the 'Common' button in the menu bar; 4 points to the 'Back' button; 5 points to the date/time display; and 6 points to a globe icon in the top right corner.

**Outline**  
This screen displays and edits the parameters of common.

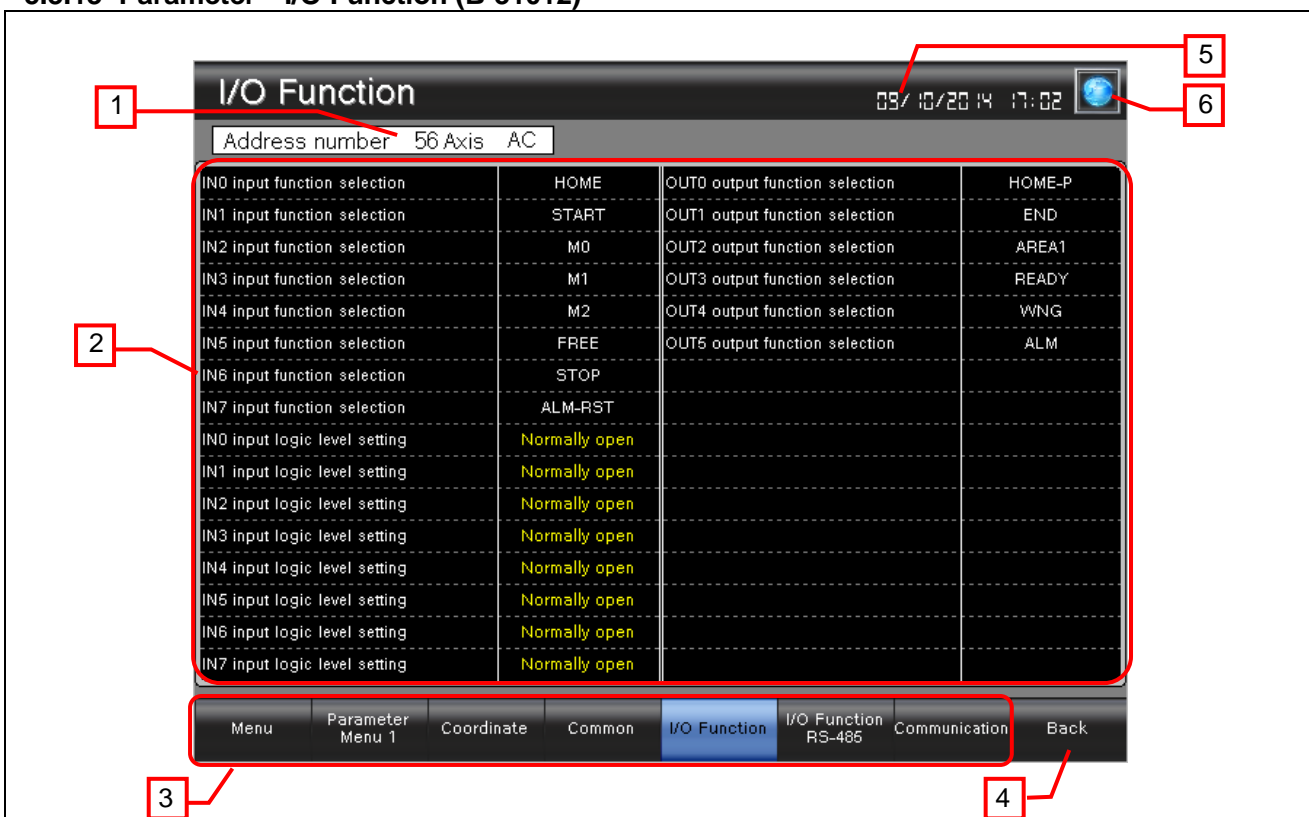
**Description**

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the common related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

**Remarks**

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If you changed the parameters of absolute backup system, please execute the configuration command. The function of the changed signal is not reflected until the configuration command is excuted. The configuration command can be executed in the [Data management] screen.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.13 Parameter I/O Function (B-31012)



#### Outline

This screen displays and edits the parameters of I/O function.

#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the I/O function related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If you changed the parameters of I/O function, please execute the configuration command. The function of the changed signal is not reflected until the configuration command is excuted. The configuration command can be executed in the [Data management] screen.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.14 Parameter I/O Function RS-485(B-31013)



#### Outline

This screen displays and edits the parameters of I/O RS-485 function.

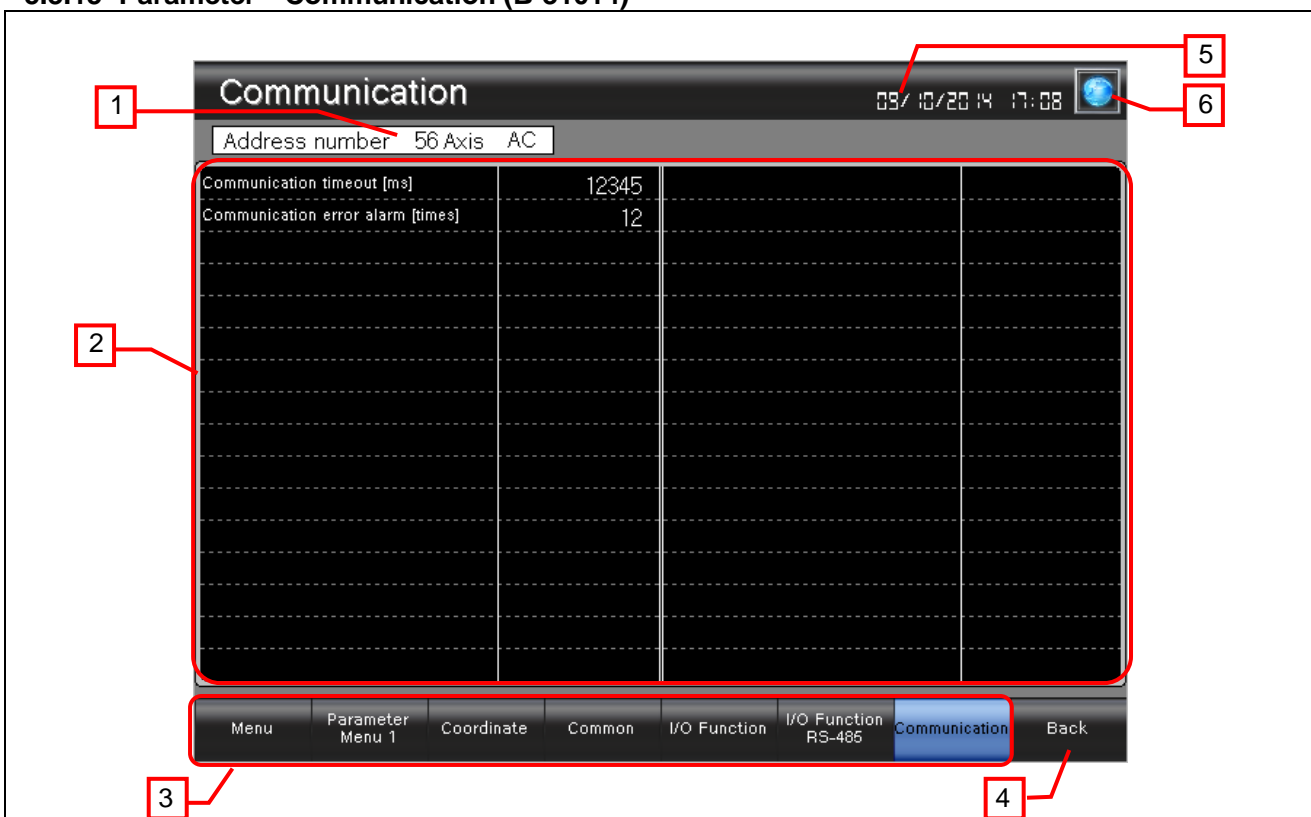
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the I/O function RS-485 related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If you changed the parameters of I/O function RS-485, please execute the configuration command. The function of the changed signal is not reflected until the configuration command is excuted. The configuration command can be executed in the [Data management] screen.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.15 Parameter Communication (B-31014)



#### Outline

This screen displays and edits the parameters of communication.

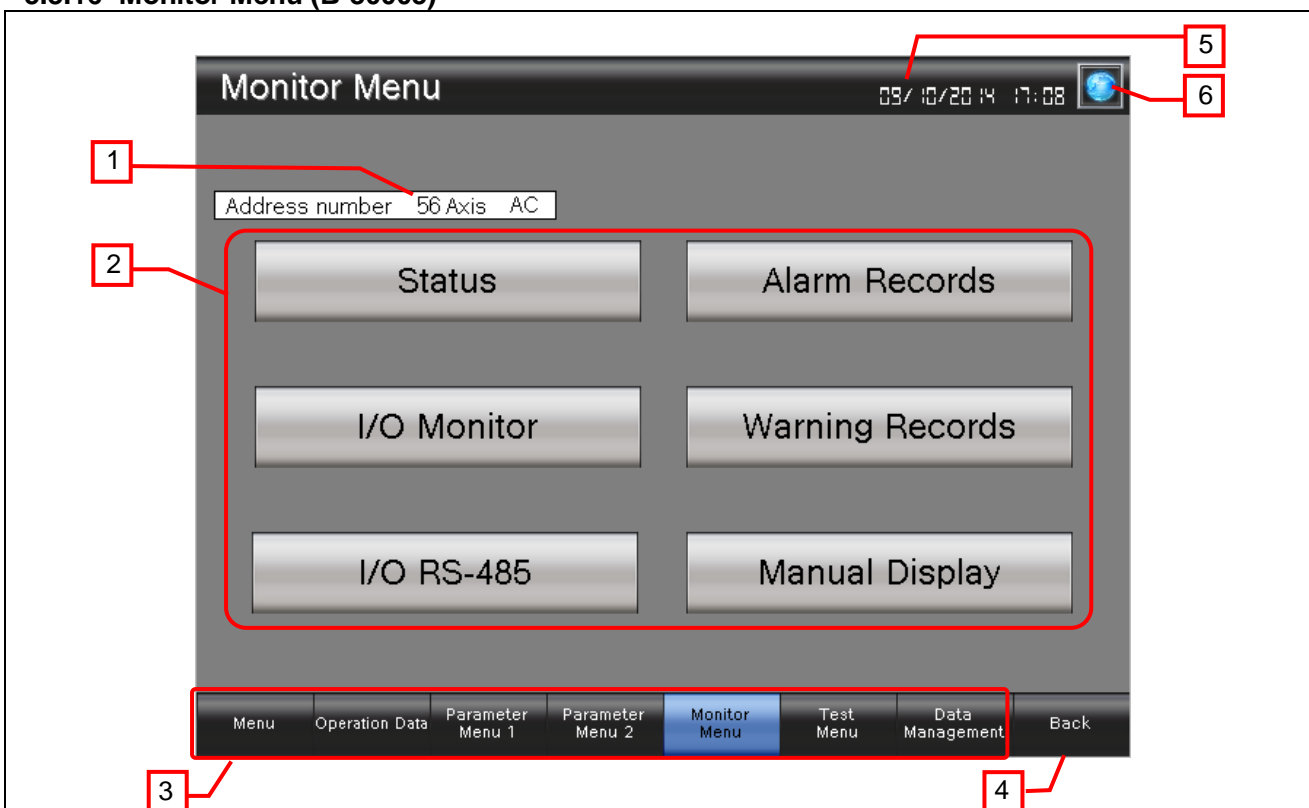
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays and edits the communication related parameters. Yellow numerical value and text indicates a initial value.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- The setting range of the parameters may vary based on the power supply type. For more details, please refer to the USER MANUAL of the driver.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.16 Monitor Menu (B-30005)



#### Outline

This is the monitor menu.

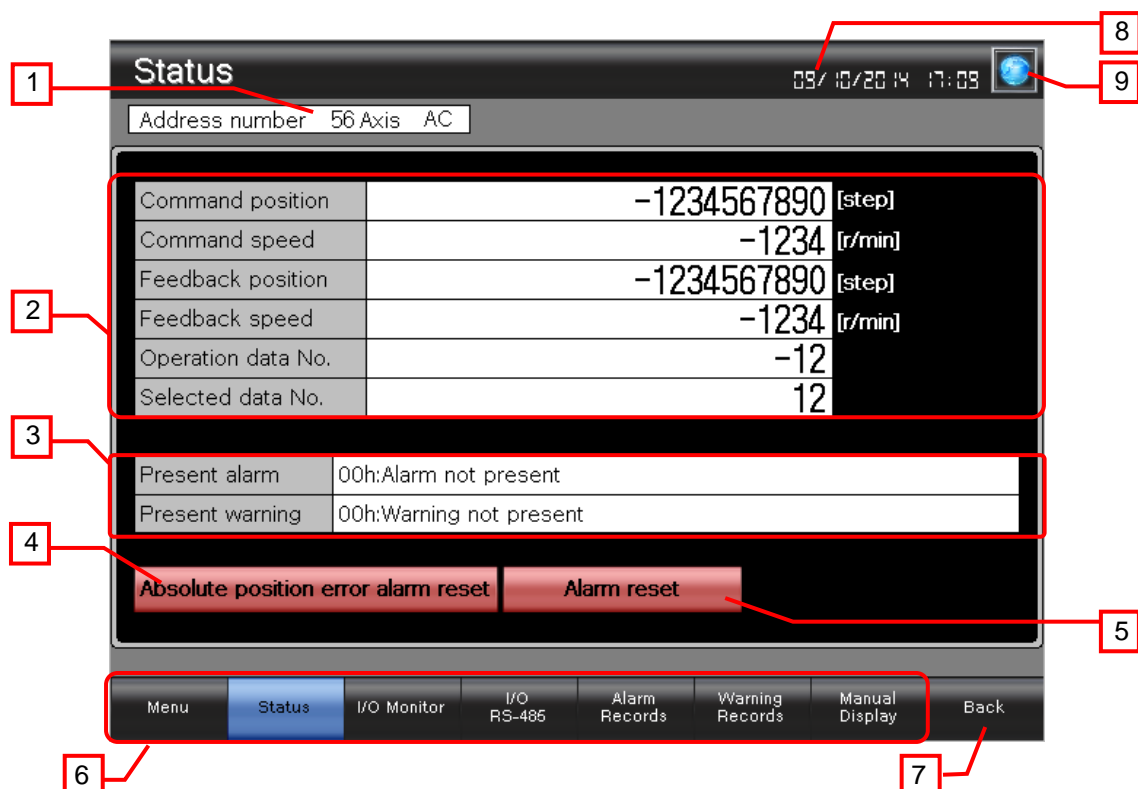
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Switches to each screen.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Switches to the previously opened screen.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.17 Monitor Status (B-31015)



#### Outline

This screen monitors the status of the motor.

#### Description

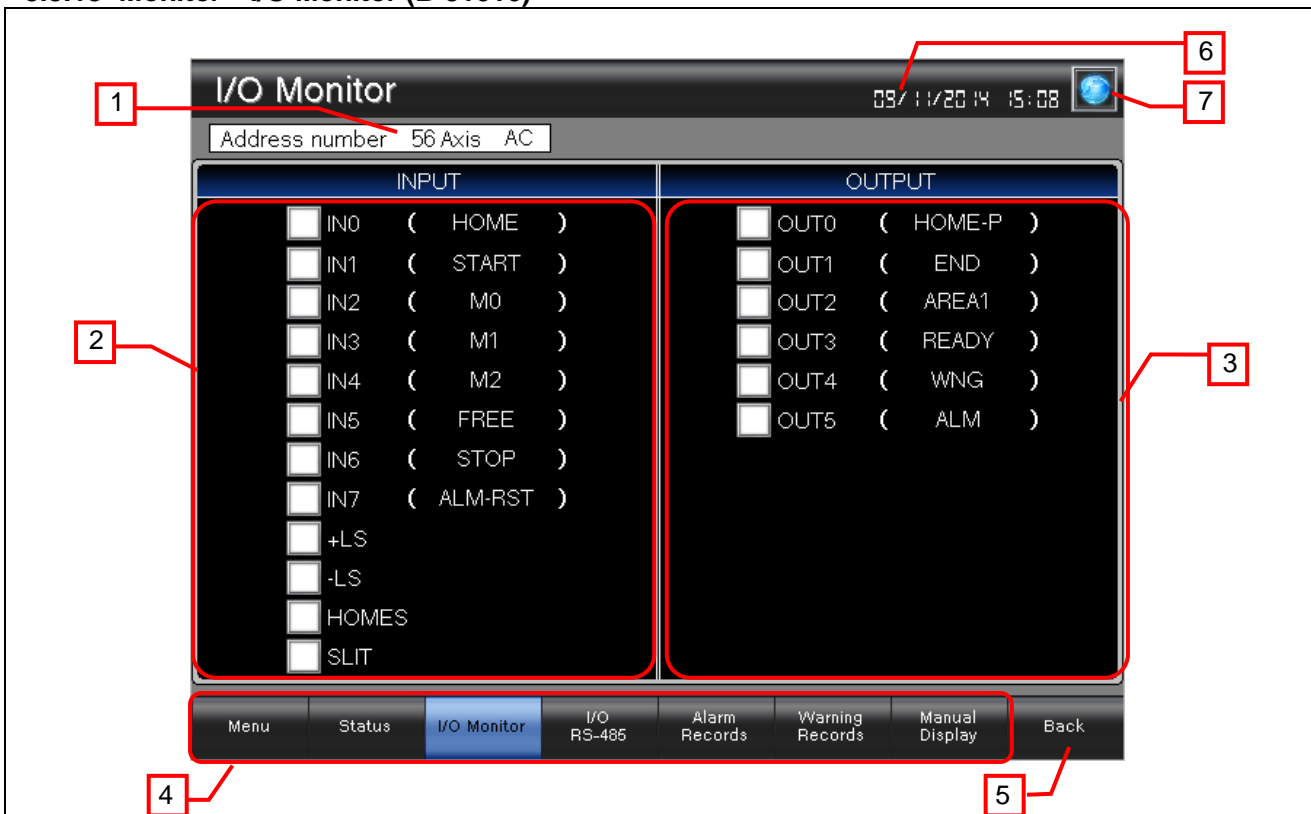
1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays the status of each item.
3. Displays the present alarm and warning.
4. Resets the absolute position error alarm. After reset, execute return-to-home operation and so on to set home position again.
5. Resets the present alarm.
6. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
7. Switches to the previously opened screen.
8. Displays the current date and time. Touch the button to open the [Clock Setting] window.
9. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.



### 5.3.18 Monitor I/O Monitor (B-31016)



#### Outline

This screen monitors I/O.

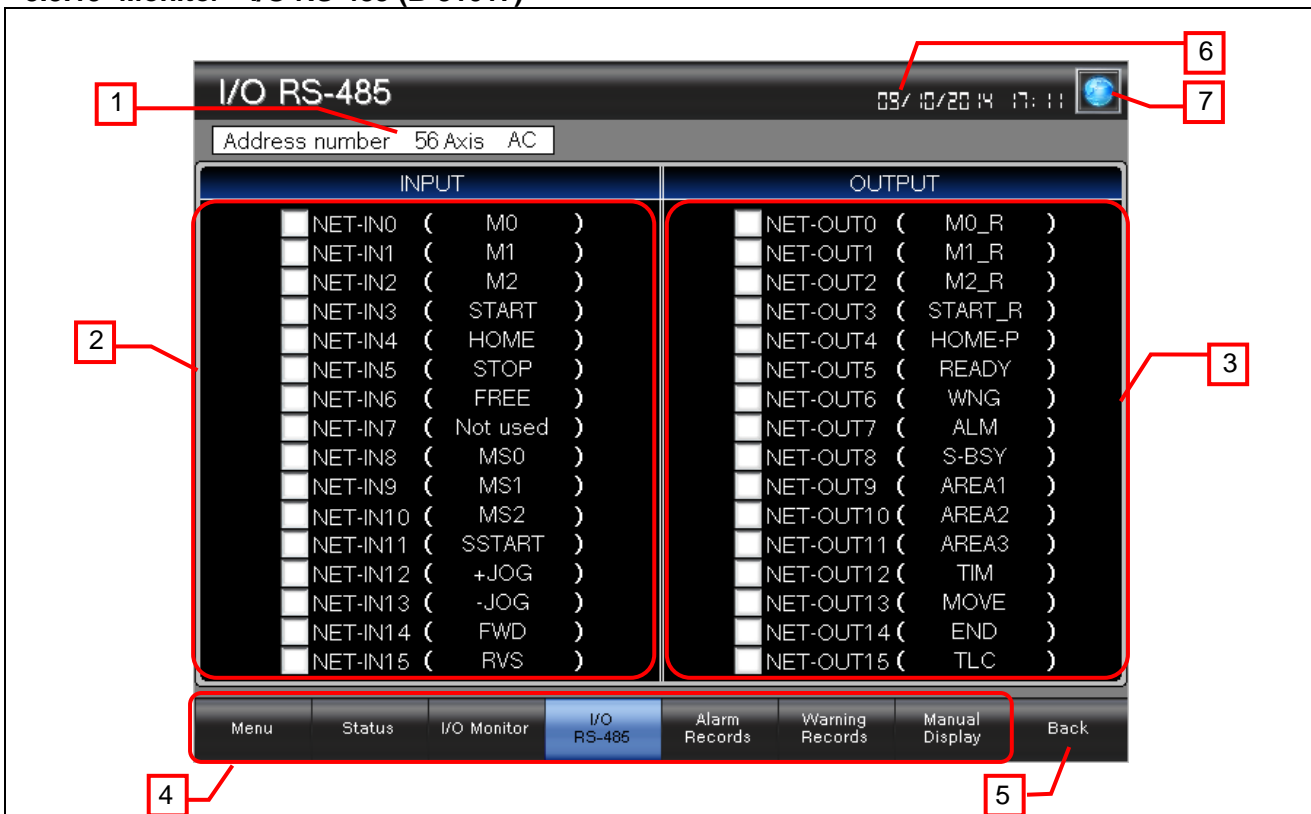
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays the status of INPUT.
3. Displays the status of OUTPUT.
4. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
5. Switches to the previously opened screen.
6. Displays the current date and time. Touch the button to open the [Clock Setting] window.
7. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.19 Monitor I/O RS-485 (B-31017)



#### Outline

This screen monitors I/O of Network communication.

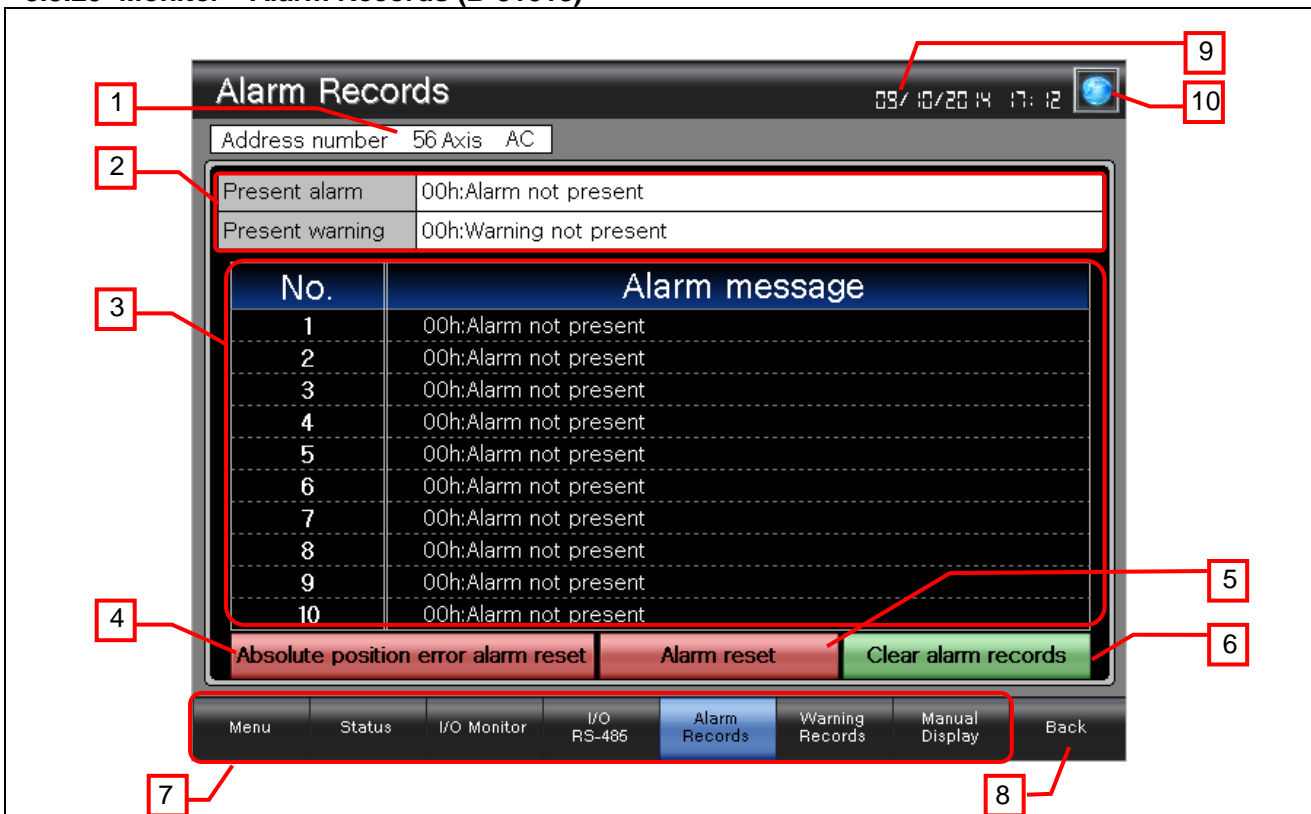
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays the status of INPUT.
3. Displays the status of OUTPUT.
4. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
5. Switches to the previously opened screen.
6. Displays the current date and time. Touch the button to open the [Clock Setting] window.
7. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.20 Monitor Alarm Records (B-31018)



#### Outline

This screen displays the alarm records.

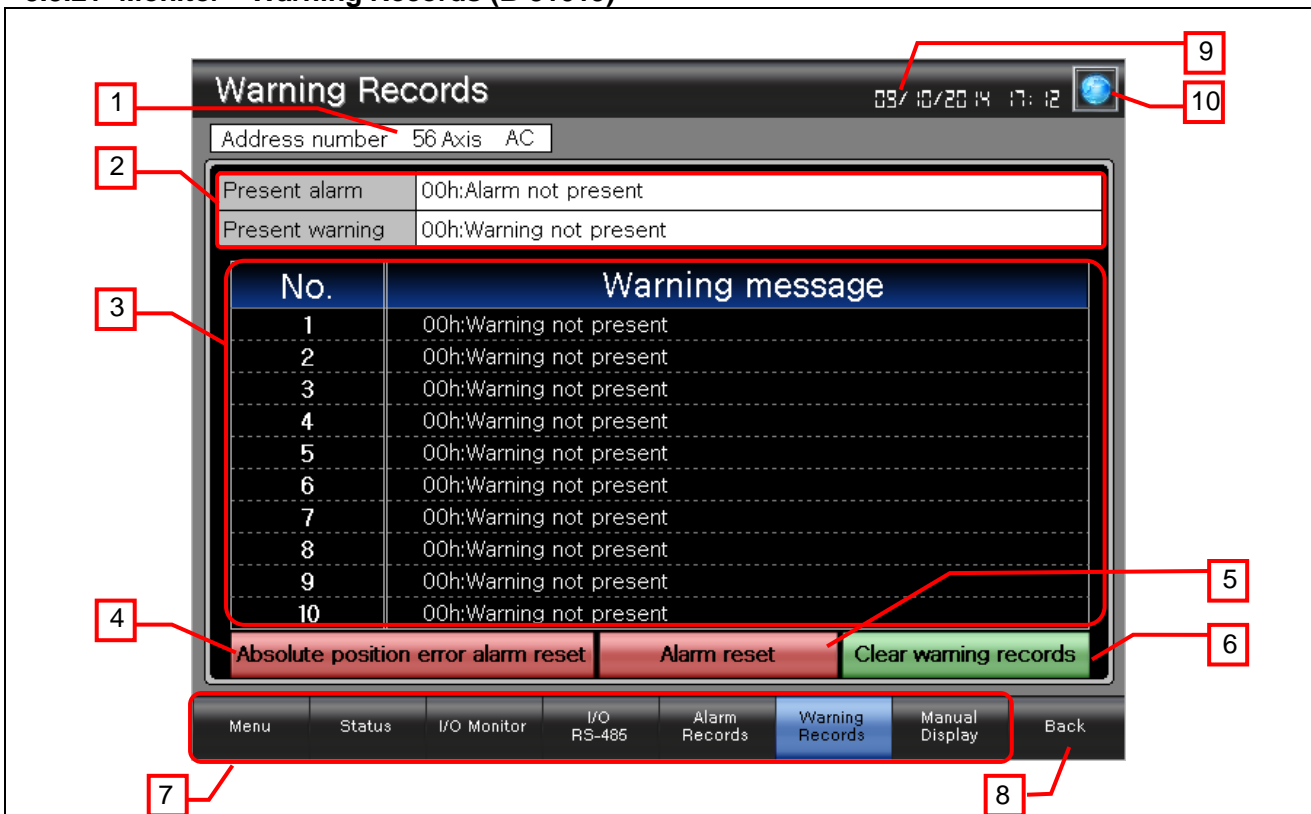
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays the present alarm and warning.
3. Displays the alarm records.
4. Resets the absolute position error alarm. After reset, execute return-to-home operation and so on to set home position again.
5. Resets the present alarm.
6. Clears the alarm records.
7. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
8. Switches to the previously opened screen.
9. Displays the current date and time. Touch the button to open the [Clock Setting] window.
10. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.21 Monitor Warning Records (B-31019)



#### Outline

This screen displays the warning records.

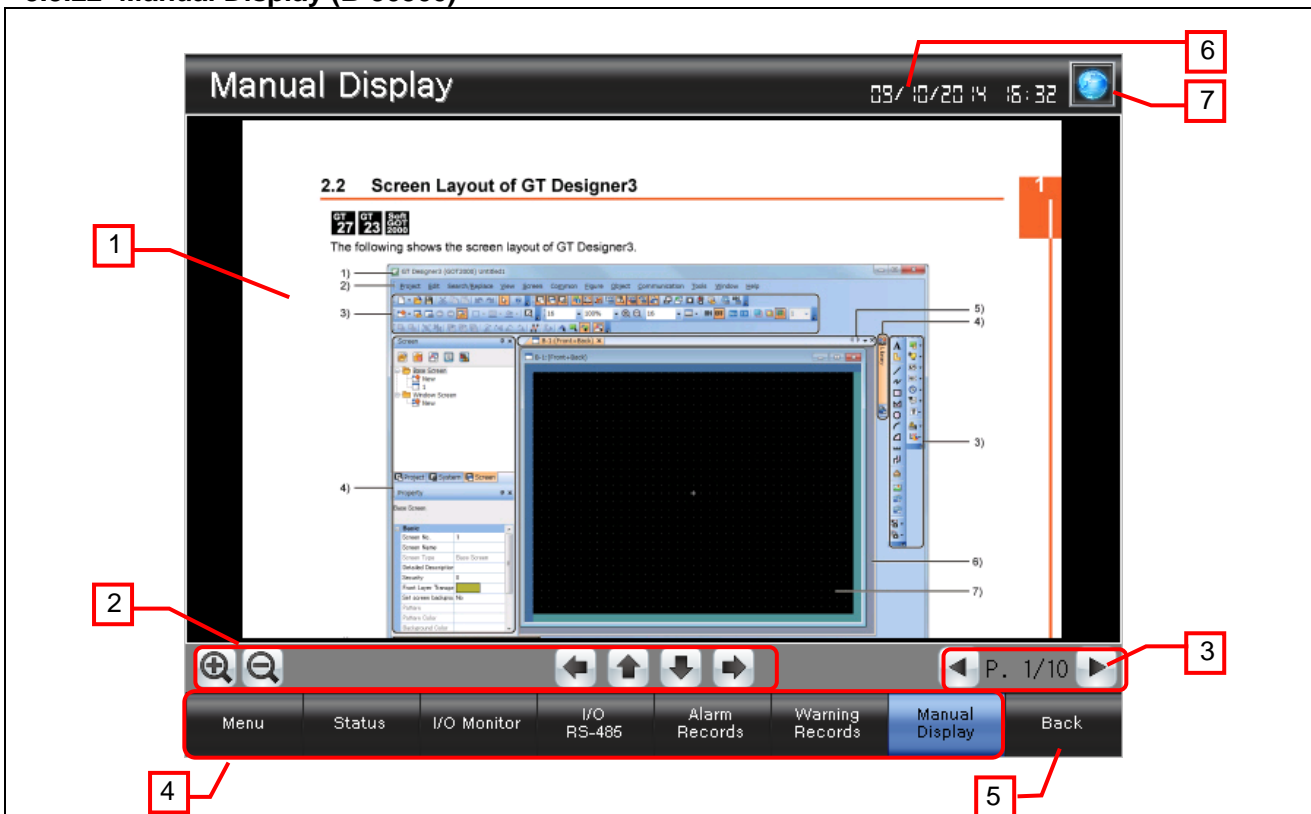
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Displays the present alarm and warning.
3. Displays the warning records.
4. Resets the absolute position error alarm. After reset, execute return-to-home operation and so on to set home position again.
5. Resets the present alarm.
6. Clears the warning records.
7. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
8. Switches to the previously opened screen.
9. Displays the current date and time. Touch the button to open the [Clock Setting] window.
10. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.22 Manual Display (B-30500)



#### Outline

This screen displays the manual of the currently displayed language.

#### Description

- Manual Display displays a document with document ID (201 to 203) according to the language. The page 1 is displayed when the screen is displayed initially. While touching the document, flicking to 8 directions will scroll the document to 8 directions. While displaying the edge of the document, flicking the document will switch pages. Pinching out and in will zoom in and out the document in 3 steps (large, middle, and small).
- These switches operate the displayed document.
  - : Enlarges or reduces the displayed document.
  - : Enlarges or reduces the displayed document.
  - : Scrolls the displayed document to the left or right.
  - : Scrolls the displayed document to the left or right.
  - : Scrolls the displayed document up or down.
  - : Scrolls the displayed document up or down.
- These switches operate the displayed document page.
  - P. 1 : Displays the page number of the displayed document. Touch the value to change the page number.
  - : Switches to the previous or next page of the displayed document.
- Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
- Switches to the previously opened screen.
- Displays the current date and time. Touch the button to open the [Clock Setting] window.
- Opens the [Language Setting] window.

## Remarks

- The language setting reflect documents for Manual display. The relation of the column No. of the comment group No., languages and document (Document ID) is shown below.

Column No. of the comment group No	Language	Document ID
1	English	201
2	Japanese	202
3	Chinese (Simplified)	203

- When GOT is started, the document page is set to No. "1" and the Document ID is set to "201" with the project script. For more details about scripts, please refer to "5.6 Script List".
- The page feed switches are set not to exceed the total number of document pages by object script. For more details about scripts, please refer to "5.6 Script List".
- The document data for the manual display should be prepared by the customers. For more details, please refer to "6. MANUAL DISPLAY".
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.23 Test Menu (B-30006)



#### Outline

This is the test menu.

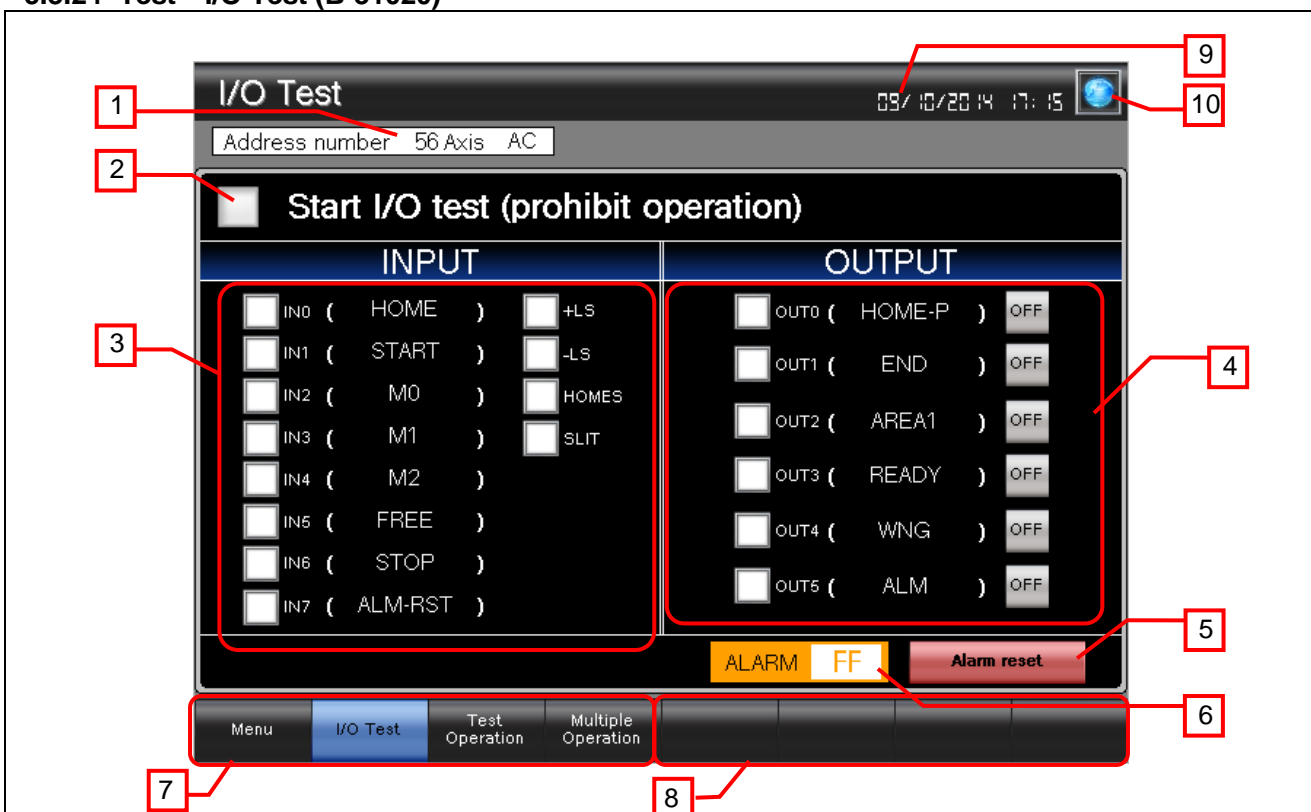
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Switches to each screen.
3. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
4. Shows unused switches for screen switching.
5. Displays the current date and time. Touch the button to open the [Clock Setting] window.
6. Opens the [Language Setting] window.

#### Remarks

- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.24 Test I/O Test (B-31020)



#### Outline

This screen checks input and test output.

#### Description

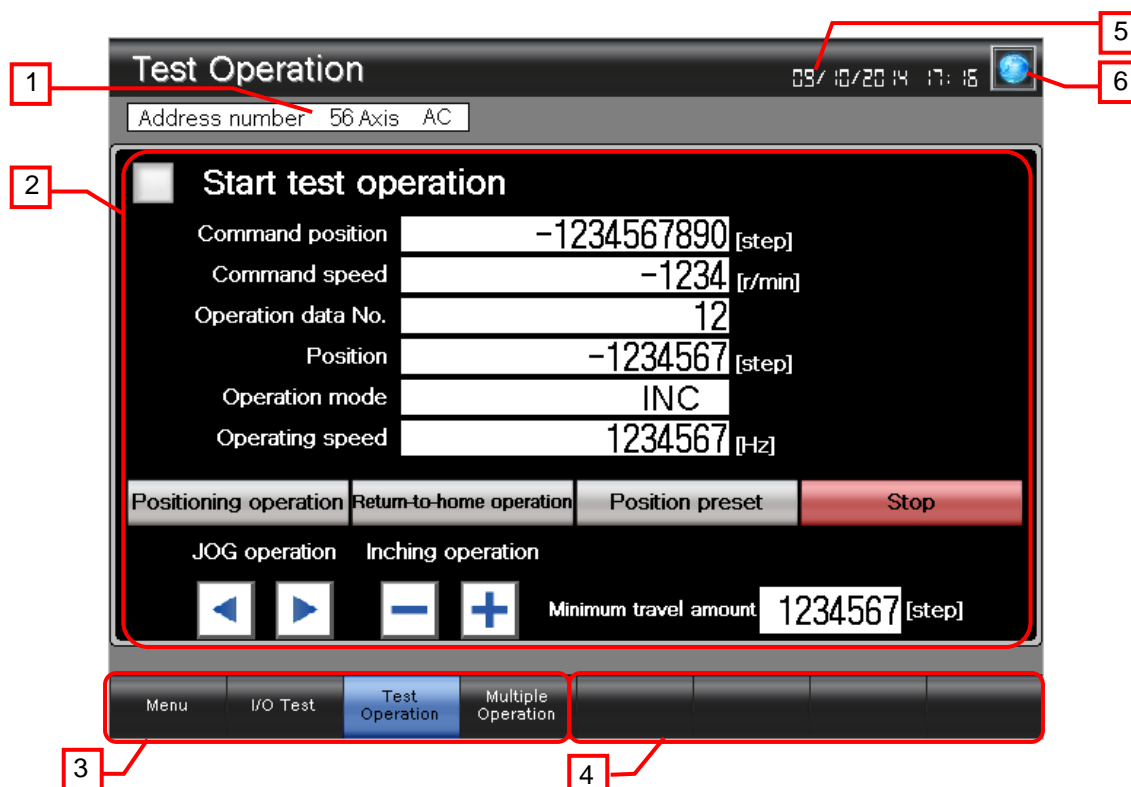
1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Touch the checkbox to start I/O test.
3. Displays the status of INPUT.
4. Displays the status of OUTPUT. Touch the switch to check the output status of OUT0 to OUT5.
5. Resets the present alarm.
6. Displays the present alarm.
7. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
8. Shows unused switches for screen switching.
9. Displays the current date and time. Touch the button to open the [Clock Setting] window.
10. Opens the [Language Setting] window.

#### Remarks

- Each time I/O test starts or completes, momentarily loses the excitation of the motor. Please pay extra attention when using vertical axis and so on. Moreover, the motor excites again after a momentary cutoff, so a position deviation occurs. If position accuracy is required, it is recommended that executing return-to-home operation immediately after I/O test is completed.
- It is not possible to switch screen or change address number when executing I/O test.
- A screen script is used for executing I/O test. For more details about scripts, please refer to "5.6 Script List".
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.



### 5.3.25 Test Operation (B-31021)



#### Outline

This screen starts test operation.

#### Description

- Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
- Touch the checkbox to start test operation.
 

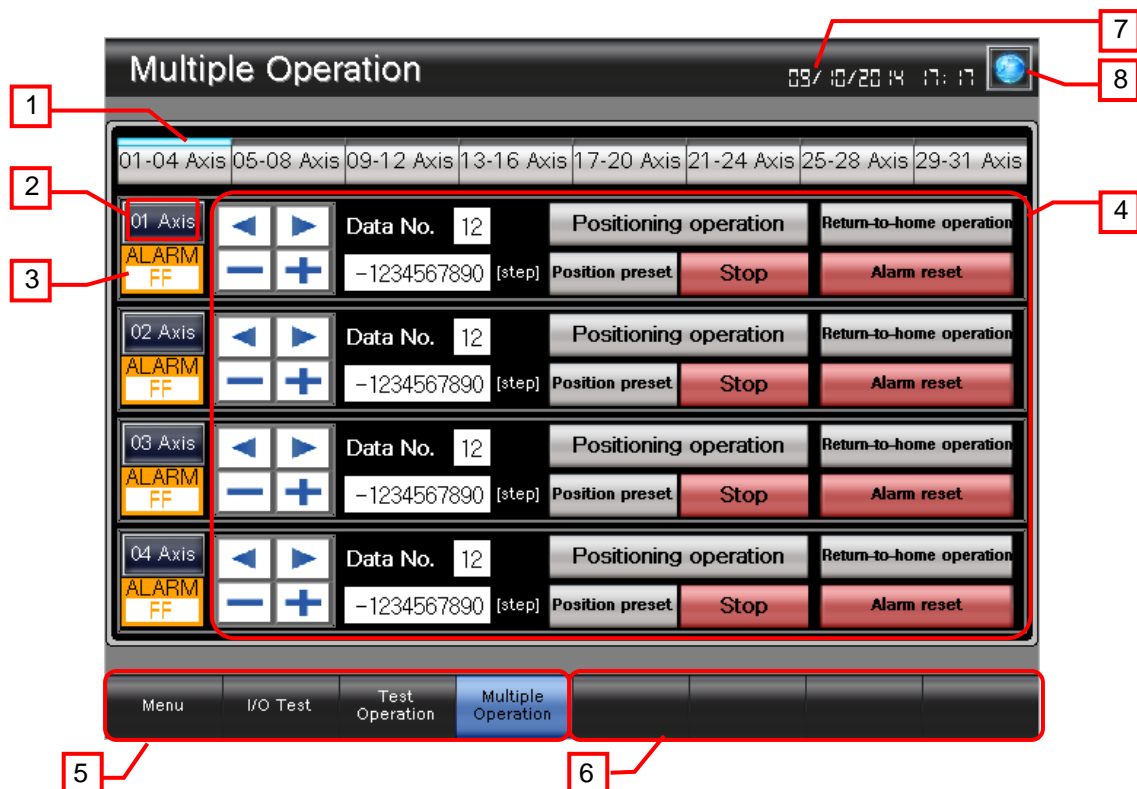
Command position	: Displays the command position.
Command speed	: Displays the command speed.
Operation data No.	: Selects the operation data No.
Position	: Displays the position.
Operation mode	: Displays the operation mode.
Operating speed	: Displays the operating speed.
Return-to-home operation	: Starts the return-to-home operation.
Positioning operation	: Executes the positioning operation according to the operation data No.
Position preset	: Sets the preset value to the command position. The preset value can be changed in "Preset position" of the [Parameter Coordinate] screen.
Stop	: Stops the operating motor.
	: Moves forward/ reverse while the button is touched. Operating speed, acceleration, deceleration according to the operation data No.
	: Adjusts the position of the motor. The travel amount of the motor equals to the minimum travel amount.

Minimum travel amount : Sets the minimum travel amount of motor.
- Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
- Shows unused switches for screen switching.
- Displays the current date and time. Touch the button to open the [Clock Setting] window.
- Opens the [Language Setting] window.

#### Remarks

- Each time test operation starts or completes, momentarily loses the excitation of the motor. Please pay extra attention when using vertical axis and so on. Moreover, the motor excites again after a momentary cutoff, so a position deviation occurs. If position accuracy is required, it is recommended that executing return-to-home operation immediately after test operation is completed.
- During test operation, acceleration and deceleration can be changed according to the operation data No. as long as acceleration/deceleration type is separate.
- It is not possible to switch screen or change address number when executing test operation.
- Some screen scripts are used for executing test operation. For more details about scripts, please refer to "5.6 Script List".
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.



### 5.3.26 Test Multiple Operation (B-31022 to 31029)



#### Outline

This screen starts test operation for a multi axis motor.

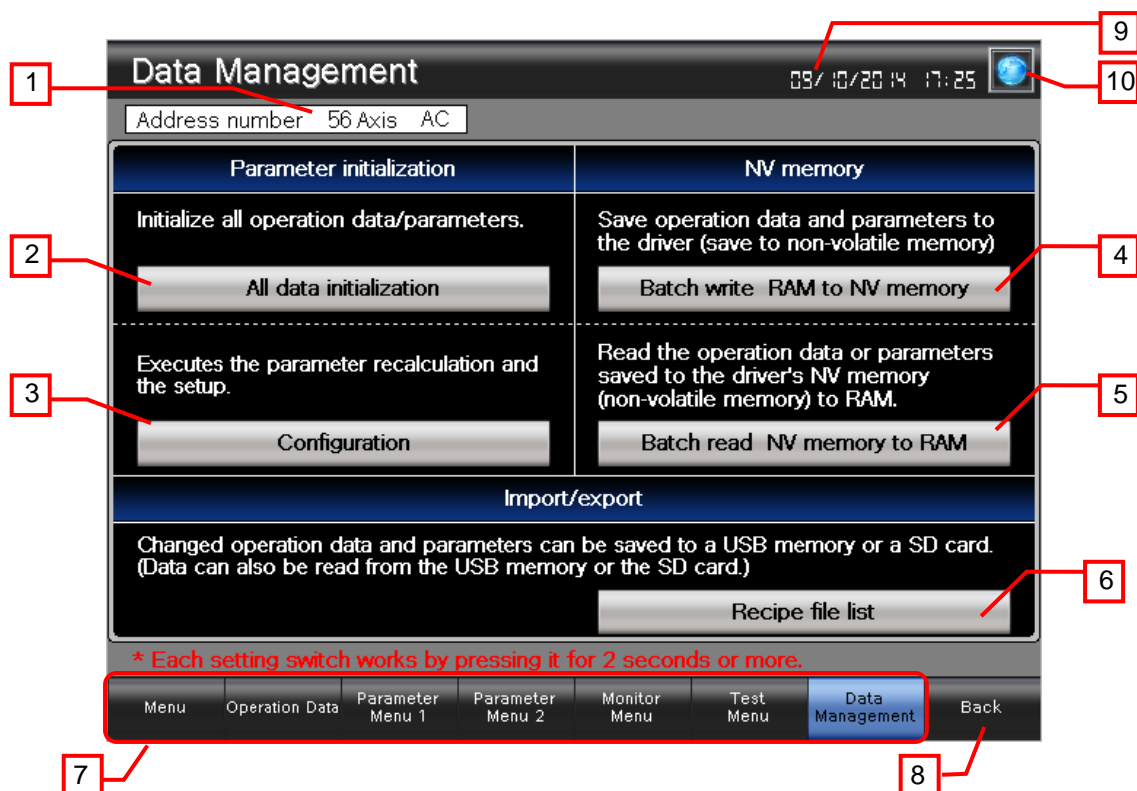
#### Description

1. Select the Axis in execute test operation of the motor.
2. Touch the switch for each Axis can be test operation.
3. Displays the present alarm.
4. Executes the test operation for each Axis.
  - Data No. : Selects the operation data No.
  -  : Moves forward/ reverse continuously while the button is touched continuously. Operating speed, acceleration, deceleration according to the operation data No.
  -  : Adjusts the position of the motor. The travel amount equals to the JOG travel amount of the operation parameter.
  - Positioning operation : Executes the positioning operation according to the operation data No.
  - Position preset : Sets the preset value to the command position. The preset value can be changed in "Preset position" of the [Parameter Coordinate] screen.
  - Stop : Stops the operating motor.
  - Return-to-home operation : Starts the Return-to-home operation.
  - Alarm reset : Resets the present alarm.
5. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
6. Shows unused switches for screen switching.
7. Displays the current date and time. Touch the button to open the [Clock Setting] window.
8. Opens the [Language Setting] window.

#### Remarks

- Each time test operation starts or completes, momentarily loses the excitation of the motor. Please pay extra attention when using vertical axis and so on. Moreover, the motor excites again after a momentary cutoff, so a position deviation occurs. If position accuracy is required, it is recommended that executing return-to-home operation immediately after test operation is completed.
- During test operation, acceleration and deceleration can be changed according to the operation data No. as long as acceleration/deceleration type is separate.
- It is not possible to switch screen or change address number when executing test operation.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.

### 5.3.27 System Data management (B-31030)



#### Outline

This screen initializes the drivers, sets up Configuration and saves/reads the operation data and the parameters.

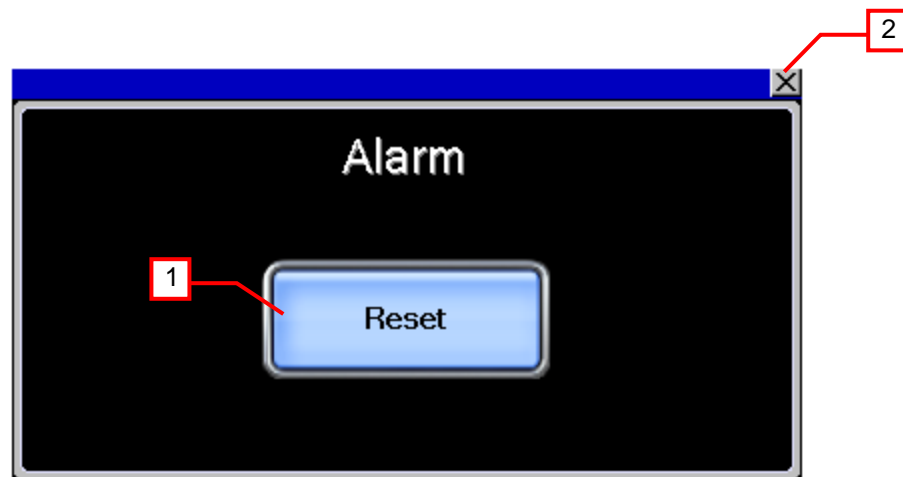
#### Description

1. Displays the address number of the driver to be monitored. Touch the button to open the [Axis Switching] window.
2. Initializes all the operation data and parameters saved in the NV memory of the driver.
3. Executes a software reset. Some parameters are not reflected until a software reset is executed.
4. Saves the operation data and parameters from the RAM to the NV memory.
5. Reads the operation data and parameters from the NV memory to the RAM.
6. Displays the [Recipe file list] window. Saves/reads the operation data and parameters to/from a USB memory or a SD card.
7. Switches to each screen. The blue switch indicates the currently displayed screen, thus selecting this switch will not switch the screen.
8. Switches to the previously opened screen.
9. Displays the current date and time. Touch the button to open the [Clock Setting] window.
10. Opens the [Language Setting] window.

#### Remarks

- Each switch works by pressing it for 2 seconds or more.
- Each time configuration command starts or completes, momentarily loses the excitation of the motor. Please pay extra attention when using vertical axis and so on. Moreover, the motor excites again after a momentary cutoff, so a position deviation occurs. If position accuracy is required, it is recommended that executing return-to-home operation immediately after configuration command is completed.
- If a system alarm occurs, the alarm message will appear at the bottom of the screen. When touching the left end of the message, the display position of the message changes in the order of upper, center, and lower. When touching the other part of the message, the [Alarm Reset] window appears.
- For more details about saving/reading operation data and parameters, please refer to "8. SAVE/READ OPERATION DATA AND PARAMETER TO/FROM A USB MEMORY OR A CF CARD".

### 5.3.28 Alarm Reset (W-30001)



#### Outline

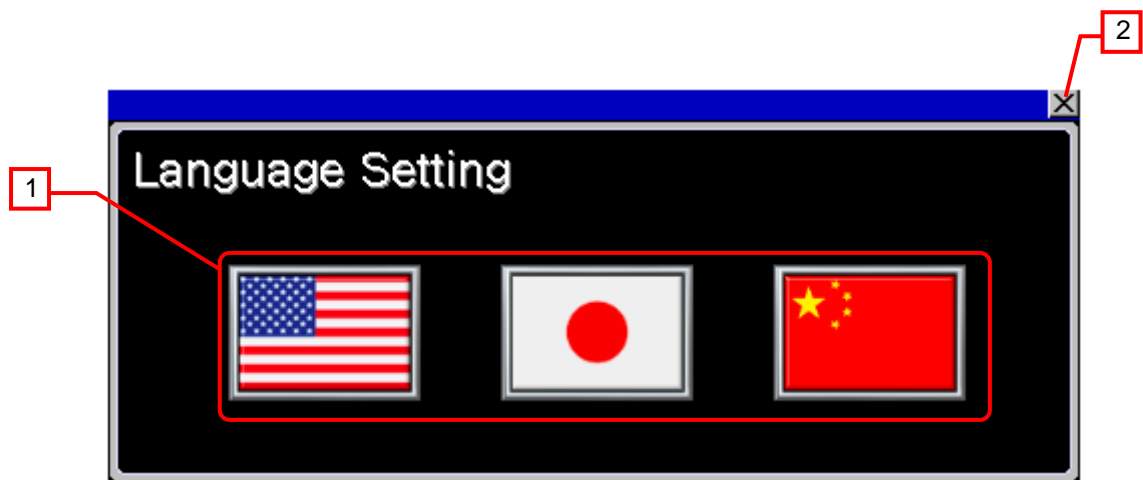
This window screen allows resetting the system alarm.

#### Description

1. Resets the system alarm, and closes the window screen after 1 second.
2. Closes the window screen.

#### Remarks

### 5.3.29 Language Setting (W-30002)



#### Outline

This window screen allows selecting the GOT language.

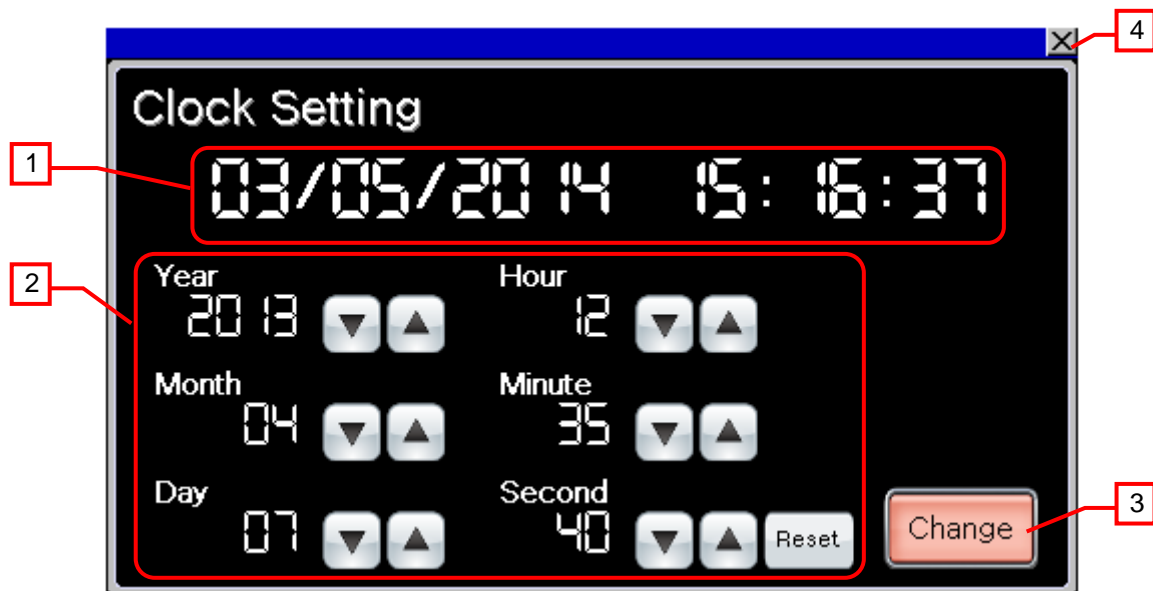
#### Description

1. Switches the language and closes the window screen.
2. Closes the window screen.

#### Remarks

- The system language and Document ID for manual display also switched corresponding to the display language.



### 5.3.30 Clock Setting (W-30003)



#### Outline

This window screen allows changing the GOT clock data.

#### Description

1. Displays the current date and time.
2. Use   switches to change the date and time. Hold down the switches to increment or decrement the value continuously. The [Reset] switch resets the seconds.
3. Applies the set date and time to the GOT clock data, and closes the window screen after 1 second.
4. Closes the window screen.

#### Remarks

- The date and time at window opening are initially set as the clock data to be newly set.
- Object scripts are set for the numerical display of the year, month, date, hour, minute and second in the clock data to be newly set. For more details about scripts, please refer to "5.6 Script List".



### 5.3.31 Axis Switching (W-30004)



#### Outline

Selects the address and the power supply type.

#### Description

1. Selects the address number of the driver to be monitored.
2. Specifies the power supply type of the driver to be monitored.
3. Closes the window screen and the settings is not reflected.
4. The settings is reflected and closes the window screen. This switch takes no action until a address number and power supply type is selected.

#### Remarks

- A screen script is used for setting the address selection and the power supply type. For more details about scripts, please refer to "5.6 Script List".

### 5.3.32 Operation Data Input (W-32001)

No. 12	
Operation mode	Incremental Absolute
Position	-1234567 [step]
Operating speed	1234567 [Hz]
Operation function	Single-motion Linked-motion Linked-motion 2 Push-motion
Push current	12.3 [%]
Dwell time	12.345 [s]
Sequential positioning	Disable Enable
Acceleration	1234.567 [ms/kHz]
Deceleration	1234.567 [ms/kHz]

Max Value	8388607
Min Value	-8388608

DEL	AC	<<	>>
7	8	9	+/-
4	5	6	
1	2	3	Enter
0	.		

Cancel Setting

#### Outline

This window screen displays and edits the operation data.

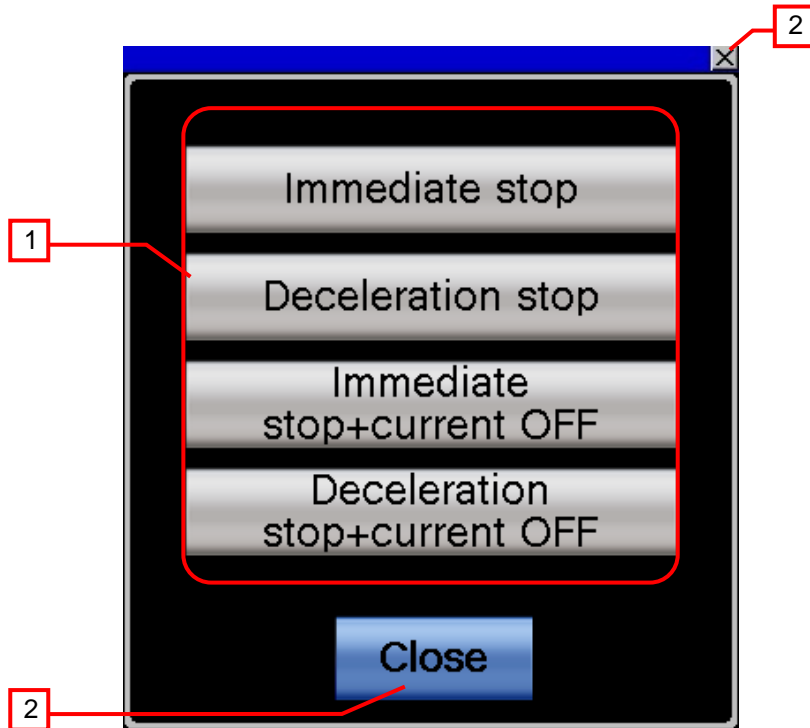
#### Description

1. Displays and edits the operation data.
2. Numeric keypad to input.
3. Closes the window screen and the edited data is not reflected.
4. The edited data is reflected and closes the window screen.

#### Remarks

- A screen script is used for read and write the operation data. For more details about scripts, please refer to "5.6 Script List".

### 5.3.33 STOP Input Action (W-32002)



#### Outline

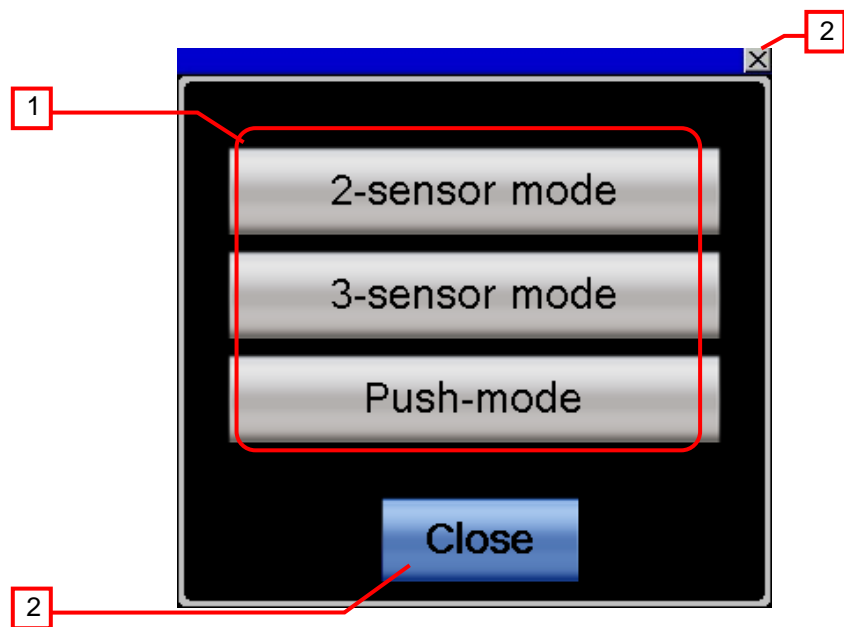
This window screen displays and sets the setting of STOP input action.

#### Description

1. Displays and sets the content of the STOP input action.
2. Closes the window screen.

#### Remarks

### 5.3.34 Home-seeking Mode (W-32003)



#### Outline

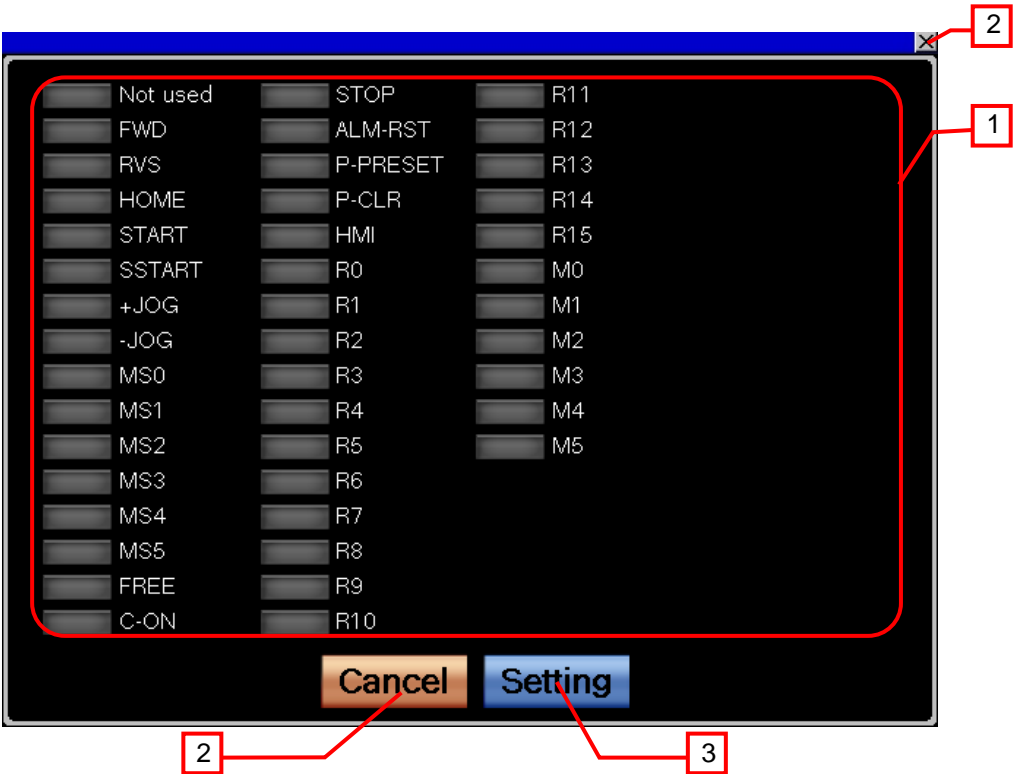
This window screen displays and sets the setting of Home-seeking mode.

#### Description

1. Displays and sets the content of the Home-seeking mode.
2. Closes the window screen.

#### Remarks

5.3.35 IN Input Function Selection (W-32004)



Outline

This window screen displays and sets the setting of IN input function selection.

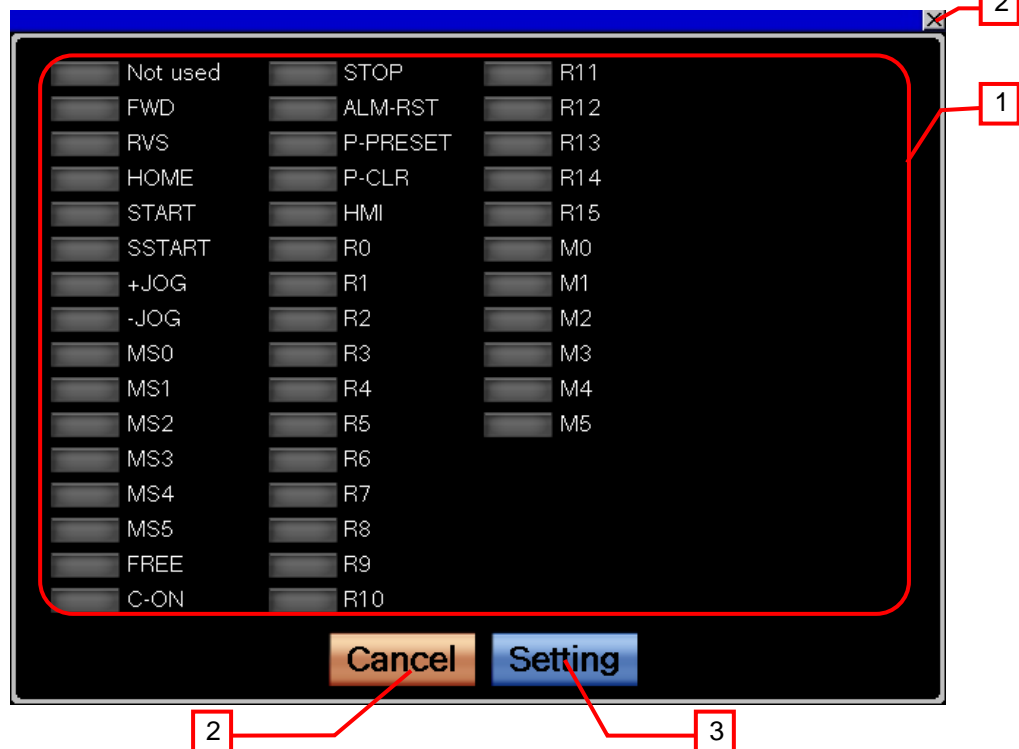
Description

1. Displays and sets the content of the IN input function selection.
2. Closes the window screen and the settings is not reflected.
3. The settings is reflected and closes the window screen.

Remarks

- A screen script is used for write the settings. For more details about scripts, please refer to “5.6 Script List”.

### 5.3.36 NET-IN Input Function Selection (W-32005)



#### Outline

This window screen displays and sets the setting of NET-IN input function selection.

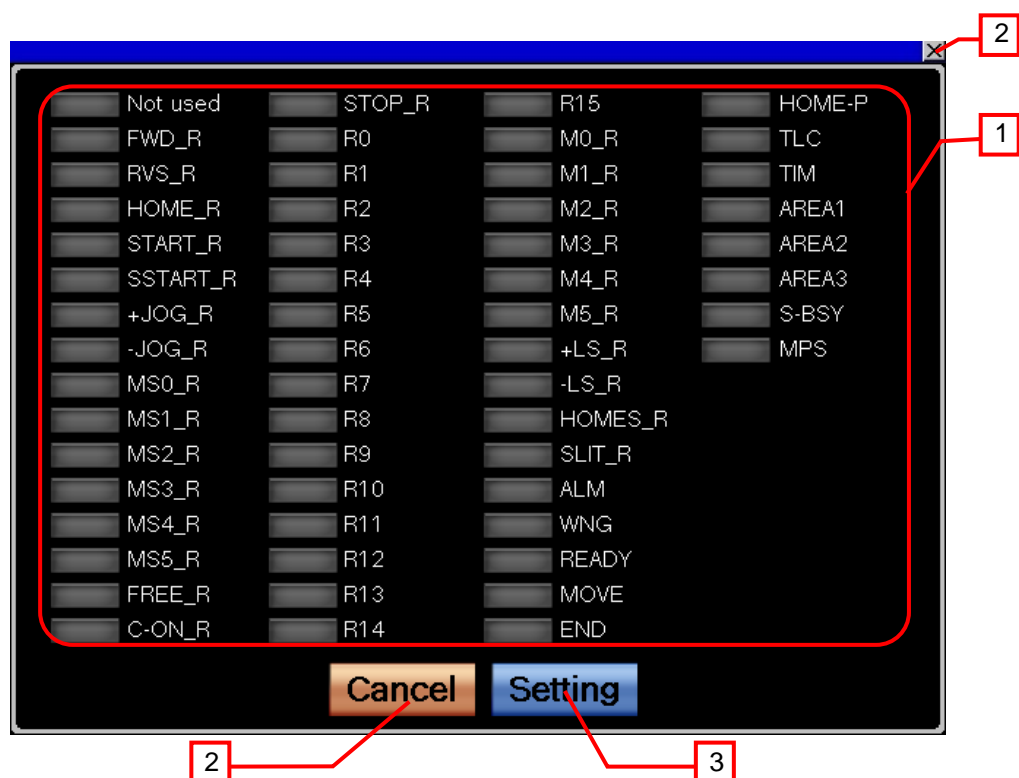
#### Description

1. Displays and sets the content of the NET-IN input function selection.
2. Closes the window screen and the settings is not reflected.
3. The settings is reflected and closes the window screen.

#### Remarks

- A screen script is used for write the settings. For more details about scripts, please refer to “5.6 Script List”.

### 5.3.37 OUT/NET-OUT Function Selection (W-32006)



#### Outline

This window screen displays and sets the setting of OUT/NET-OUT function selection.

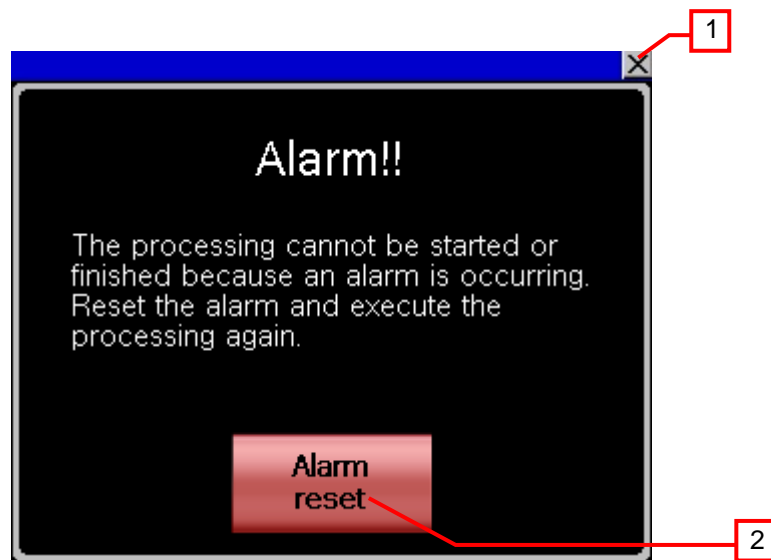
#### Description

1. Displays and sets the content of the OUT/NET-OUT function selection.
2. Closes the window screen and the settings is not reflected.
3. The settings is reflected and closes the window screen.

#### Remarks

- MPS is only valid for AC power supply type.
- A screen script is used for write the settings. For more details about scripts, please refer to “5.6 Script List”.

### 5.3.38 Alarm Occurring Confirmation (W-32007)



#### Outline

If a teaching or test operation is executed when there is any alarm present, this window appears.

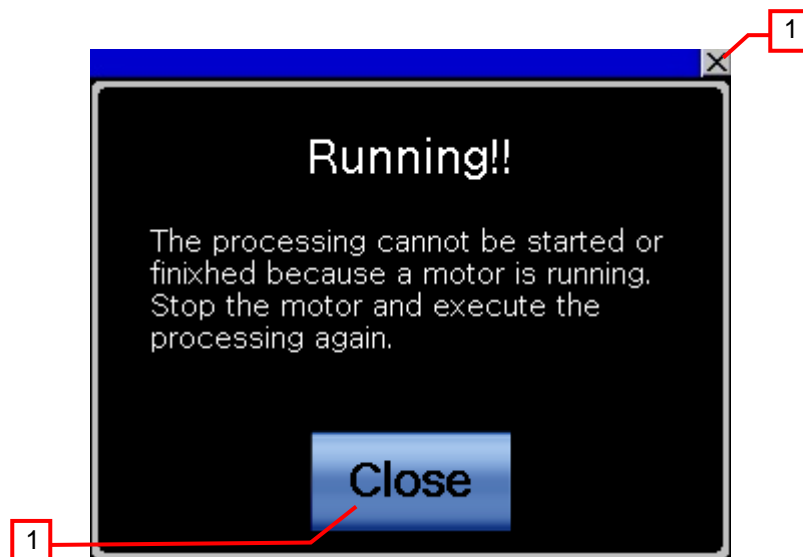
#### Description

1. Closes the window screen.
2. Resets the present alarm.

#### Remarks



### 5.3.39 Motor Running Confirmation (W-32008)



#### Outline

If a teaching or test operation is executed when the motor is running, this window appears.

#### Description

1. Closes the window screen.

#### Remarks

## 5.4 Device List

Some of the devices specified to the on-screen switches and lamps, etc., are also used for common settings of functions such as scripts. Using [Batch Edit] is recommended to change these devices in a batch. For more details about using [Batch Edit], please refer to the "GT Designer3 (GOT2000) Help".

### 5.4.1 Devices of the controller

Type	Device No.	Application
Bit	400126.b0	Driver input command [M0]
	400126.b1	Driver input command [M1]
	400126.b2	Driver input command [M2]
	400126.b3	Driver input command [START]
	400126.b4	Driver input command [HOME]
	400126.b5	Driver input command [STOP]
	400126.b6	Driver input command [FREE]
	400126.b7	Driver input command [Not used]
	400126.b8	Driver input command [MS0]
	400126.b9	Driver input command [MS1]
	400126.b10	Driver input command [MS2]
	400126.b11	Driver input command [SSTART]
	400126.b12	Driver input command [+JOG]
	400126.b13	Driver input command [-JOG]
	400126.b14	Driver input command [FWD]
	400126.b15	Driver input command [RVS]
	400128.b0	Driver output command [M0_R]
	400128.b1	Driver output command [M1_R]
	400128.b2	Driver output command [M2_R]
	400128.b3	Driver output command [START_R]
	400128.b4	Driver output command [HOME-P]
	400128.b5	Driver output command [READY]
	400128.b6	Driver output command [WNG]
	400128.b7	Driver output command [ALM]
	400128.b8	Driver output command [S-BSY]
	400128.b9	Driver output command [AREA1]
	400128.b10	Driver output command [AREA2]
	400128.b11	Driver output command [AREA3]
	400128.b12	Driver output command [TIM]
	400128.b13	Driver output command [MOVE]
	400128.b14	Driver output command [END]
	400128.b15	Driver output command [TLC]
	400213.b0	Direct I/O and electromagnetic brake status [OUT0]
	400213.b1	Direct I/O and electromagnetic brake status [OUT1]
	400213.b2	Direct I/O and electromagnetic brake status [OUT2]
	400213.b3	Direct I/O and electromagnetic brake status [OUT3]
	400213.b4	Direct I/O and electromagnetic brake status [OUT4]
	400213.b5	Direct I/O and electromagnetic brake status [OUT5]
	400214.b0	Direct I/O and electromagnetic brake status [+LS]
	400214.b1	Direct I/O and electromagnetic brake status [-LS]
	400214.b2	Direct I/O and electromagnetic brake status [HOMES]
	400214.b3	Direct I/O and electromagnetic brake status [SLIT]
	400214.b6	Direct I/O and electromagnetic brake status [IN0]
	400214.b7	Direct I/O and electromagnetic brake status [IN1]
	400214.b8	Direct I/O and electromagnetic brake status [IN2]
	400214.b9	Direct I/O and electromagnetic brake status [IN3]
	400214.b10	Direct I/O and electromagnetic brake status [IN4]
	400214.b11	Direct I/O and electromagnetic brake status [IN5]
	400214.b12	Direct I/O and electromagnetic brake status [IN6]
	400214.b13	Direct I/O and electromagnetic brake status [IN7]
	400386.b0	Reset alarm
	400388.b0	Absolute position error alarm reset
	400390.b0	Clear alarm records

Type	Device No.	Application
Bit	400392.b0	Clear warning records
	400396.b0	P-PRESET execute
	400398.b0	Configuration execute
	400400.b0	All data initialization
	400402.b0	Batch NV memory read
	400404.b0	Batch NV memory read
	400516.b0	Hardware overtravel
	400518.b0	Overtravel action
	400538.b0	LS logic level
	400540.b0	HOMES logiclevel
	400542.b0	SLIT logiclevel
	400654.b0	Acceleration/ deceleration type
	400656.b0	Acceleration/ deceleration unit
	400716.b0	Starting direction of home-seeking
	400718.b0	SLIT detection with home-seeking
	400720.b0	TIM signal detection with home-seeking
	400778.b0	Return-to-home incomplete alarm
	400902.b0	Motor rotation direction
	400904.b0	Software overtravel
	400912.b0	Wrap setting
	400962.b0	Data setter speed display
	400964.b0	Data setter edit
	400966.b0	Absolute-position backup system
	404110.b0	HOME-P function selection
	404130.b0	Filter selection
	404136.b0	Control mode
	404138.b0	Smooth driver
	404162.b0	Automatic return action
	404386.b0	IN0 input logic level setting
	404388.b0	IN1 input logic level setting
	404390.b0	IN2 input logic level setting
	404392.b0	IN3 input logic level setting
	404394.b0	IN4 input logic level setting
	404396.b0	IN5 input logic level setting
	404398.b0	IN6 input logic level setting
	404400.b0	IN7 input logic level setting
Word	400126	Driver input command (Lower)
	400129	Present alarm (Upper)
	400130	Present alarm (Lower)
	400132+2n (n=0 to 9)	Alarm record 1 to 10 (Upper)
	400152	Present warning (Lower)
	400154+2n (n=0 to 9)	Warning record 1 to 10 (Upper)
	400195	Present selected Data No. (Upper)
	400196	Present selected Data No. (Lower)
	400197	Present operation Data No. (Upper)
	400199	Command position (Upper)
	400201	Command speed (Upper)
	400205	Feedback position (Upper)
	400207	Feedback speed (Upper)
	400395	P-PRESET execute (Upper)
	400397	Configuration (Upper)
	400513	STOP input action (Upper)
	400514	STOP input action (Lower)
	400515	Hardware overtravel (Upper)
	400517	Overtravel action (Upper)
	400519	Positioning completion signal range (Upper)
	400521	Positioning completion signal offset (Upper)
	400523	AREA1 positive direction position (Upper)
	400525	AREA1 negative direction position (Upper)
	400527	AREA2 positive direction position (Upper)

Type	Device No.	Application
Word	400529	AREA2 negative direction position (Upper)
	400531	AREA3 positive direction position (Upper)
	400533	AREA3 negative direction position (Upper)
	400535	Minimum ON time for MOVE output (Upper)
	400537	LS logic level (Upper)
	400539	HOMES logic level (Upper)
	400541	SLIT logic level (Upper)
	400577	RUN current (Upper)
	400579	STOP current (Upper)
	400581	Position loop gain (Upper)
	400583	Speed loop gain (Upper)
	400585	Speed loop integral time constant (Upper)
	400587	Speed filter (Upper)
	400589	Moving average time (Upper)
	400641	Common acceleration (Upper)
	400643	Common deceleration (Upper)
	400645	Starting speed (Upper)
	400647	JOG operating speed (Upper)
	400649	Acceleration/deceleration rate of JOG (Upper)
	400651	JOG starting speed (Upper)
	400653	Acceleration/deceleration type (Upper)
	400655	Acceleration/deceleration unit (Upper)
	400656	Acceleration/deceleration unit (Lower)
	400705	Home-seeking mode (Upper)
	400706	Home-seeking mode (Lower)
	400707	Operating speed of home-seeking (Upper)
	400709	Acceleration of home-seeking (Upper)
	400711	Starting speed of home-seeking (Upper)
	400713	Position offset of home-seeking (Upper)
	400715	Starting direction of home-seeking (Upper)
	400717	SLIT detection with home-seeking (Upper)
	400719	TIM signal detection with home-seeking (Upper)
	400721	Operating current of push-motion home-seeking (Upper)
	400769	Overload alarm (Upper)
	400771	Overflow rotation alarm during current on (Upper)
	400777	Return-to-home incomplete alarm (Upper)
	400833	Overheat warning (Upper)
	400835	Overload warning (Upper)
	400837	Overspeed warning (Upper)
	400839	Overvoltage warning (Upper)
	400841	Undervoltage warning (Upper)
	400843	Overflow rotation warning during current on (Upper)
	400897	Electronic gear A (Upper)
	400899	Electronic gear B (Upper)
	400901	Motor rotation direction (Upper)
	400903	Software overtravel (Upper)
	400905	Positive software Limit (Upper)
	400907	Negative software Limit (Upper)
	400909	Preset position (Upper)
	400911	Wrap setting (Upper)
	400913	Wrap setting range (Upper)
	400961	Data setter speed display (Upper)
	400963	Data setter edit (Upper)
	400965	Absolute-position backup system (Upper)
	401025+2n (n=0 to 63)	Position No.0 to No.63 (Upper)
	401153+2n (n=0 to 63)	Operating speed No.0 to No.63 (Upper)
	401281+2n (n=0 to 63)	Operation mode No.0 to No.63 (Upper)
	401282+2n (n=0 to 63)	Operation mode No.0 to No.63 (Lower)
	401409+2n (n=0 to 63)	Operation function No.0 to No.63 (Upper)
	401410+2n (n=0 to 63)	Operation function No.0 to No.63 (Lower)

Type	Device No.	Application
Word	401537+2n (n=0 to 63)	Acceleration No.0 to No.63 (Upper)
	401665+2n (n=0 to 63)	Deceleration No.0 to No.63 (Upper)
	401793+2n (n=0 to 63)	Push current No.0 to No.63 (Upper)
	401921+2n (n=0 to 63)	Sequential positioning No.0 to No. 63 (Upper)
	401922+2n (n=0 to 63)	Sequential positioning No.0 to No.63 (Lower)
	402049+2n (n=0 to 63)	Dwell time No.0 to No.63 (Upper)
	404097+2n (n=0 to 5)	MS0 operation No. selection (Upper) to MS5 operation No. selection (Upper)
	404109	HOME-P function selection (Upper)
	404129	Filter selection (Upper)
	404131	Speed error gain 1(Upper)
	404133	Speed error gain 2(Upper)
	404135	Control mode (Upper)
	404137	Smooth driver (Upper)
	404161	Automatic return action (Upper)
	404163	Operating speed of automatic return (Upper)
	404165	Acceleration/deceleration of automatic return (Upper)
	404167	Starting speed of automatic return (Upper)
	404169	JOG operation (Upper)
	404225	Overflow rotation alarm during current off (Upper)
	404353+2n (n=0 to 7)	IN0 input function selection (Upper) to IN7 input function selection (Upper)
	404354+2n (n=0 to 7)	IN0 input function selection (Lower) to IN7 input function selection (Lower)
	404385+2n (n=0 to 7)	IN0 input logic level setting (Upper) to IN7 input logic level setting (Upper)
	404417+2n (n=0 to 5)	OUT0 output function selection (Upper) to OUT5 output function selection (Upper)
	404418+2n (n=0 to 5)	OUT0 output function selection (Lower) to OUT5 output function selection (Lower)
	404449+2n (n=0 to 15)	NET-IN0 input function selection (Upper) to NET-IN15 input function selection (Upper)
	404450+2n (n=0 to 15)	NET-IN0 input function selection (Lower) to NET-IN15 input function selection (Lower)
	404481+2n (n=0 to 15)	NET-OUT0 output function selection (Upper) to NET-OUT15 output function selection (Upper)
	404482+2n (n=0 to 15)	NET-OUT0 output function selection (Lower) to NET-OUT15 output function selection (Lower)
	404609	Communication timeout (Upper)
	404611	Communication error alarm (Lower)

#### 5.4.2 GOT internal devices

Type	Device No.	Application
Bit	GB40	Script trigger
	GB61000	Status flag
	GB61001	Address number selection status flag
	GB61002	Power supply type status flag
	GB61010	Power supply type distinguishment flag
	GB62000	Display operation data scroll up
	GB62001	Display operation data scroll down
	GB62002	Read trigger of operation data input
	GB62003	Write trigger of operation data input
	GB62004	Write trigger of I/O function selection
	GB62005	OUT/NET-OUT distinguishment bit
	GB62006	Overlap window display flag
	GB62007	Interlock flag for screen transition prohibition
	GB62008	Interlock flag for multiple operation
	GB62009	Multiple start/completion flag
	GB62010	Check trigger before start/completion
	GB62011	Alarm status check flag
	GB62012	Operating check flag
	GB62013	Trigger for operation ready to start
	GB62014	Operation preparing flag
	GB62015	Touch panel operating flag
	GB62016	Trigger for start of operation completion
	GB62017	Operation completion flag
	GB62018	I/O test flag
	GB62019	Position set trigger
	GB62020+10n (n=0 to 30)	Check trigger before start/completion (Axis No.1 to Axis No.31)
	GB62021+10n (n=0 to 30)	Alarm status check flag (Axis No.1 to Axis No.31)
	GB62022+10n (n=0 to 30)	Operating check flag (Axis No.1 to Axis No.31)
	GB62023+10n (n=0 to 30)	Trigger for operation ready to start (Axis No.1 to Axis No.31)
	GB62024+10n (n=0 to 30)	Operation preparing flag (Axis No.1 to Axis No.31)
	GB62025+10n (n=0 to 30)	Touch panel operating flag (Axis No.1 to Axis No.31)
	GB62026+10n (n=0 to 30)	Trigger for start of operation completion (Axis No.1 to Axis No.31)
	GB62027+10n (n=0 to 30)	Operation completion flag (Axis No.1 to Axis No.31)
	GD60031.b0	Alarm reset
	GD61020.b0 to b5	Display operation number No. of operation data input screen
	GD61200+100n.b0 (n=0 to 30)	Display operation number No. (Axis No.1 to Axis No.31)
	GD61200+100n.b1 (n=0 to 30)	Display operation number No. (Axis No.1 to Axis No.31)
	GD61200+100n.b2 (n=0 to 30)	Display operation number No. (Axis No.1 to Axis No.31)
	GD61200+100n.b3 (n=0 to 30)	Display operation number No. (Axis No.1 to Axis No.31)
	GD61200+100n.b4 (n=0 to 30)	Display operation number No. (Axis No.1 to Axis No.31)
	GD61200+100n.b5 (n=0 to 30)	Display operation number No. (Axis No.1 to Axis No.31)
	GS281.b0 to GS282.b15	Faulty station information (CH1)
	GS512.b0	Time change signal
	GS531.b0 to GS532.b15	Monitor station disconnection (CH1)
Word	GD60000	Base screen switching
	GD60001	Overlap window 1 screen switching
	GD60004	Overlap window 2 screen switching
	GD60021	Language switching
	GD60022	System language switching
	GD60031、GD60041	System information
	GD60042	Current cursor display user ID

Type	Device No.	Application
Word	GD60080 to GD60082	Document display
	GD61000	Address number input
	GD61001 to GD61008	Display the number of the operation data No. (Row 0 to Row 7)
	GD61010	Scroll up/down counter value
	GD61011	Offset value of Modbus address
	GD61012	Touch position information of operation data No.
	GD61015	Command position when teaching operates、offset value of Modbus address in minimum travel amount
	GD61020	Display the operation number No. of operation data input screen
	GD61021	Operation mode of operation data input screen
	GD61022	Position of operation data input screen
	GD61024	Operating speed of operation data input screen
	GD61026	Operation function of operation data input screen
	GD61027	Push current of operation data input screen
	GD61029	Dwell time of operation data input screen
	GD61031	Sequential positioning of operation data input screen
	GD61032	Acceleration of operation data input screen
	GD61034	Deceleration of operation data input screen
	GD61040	The signal number of IN input
	GD61041	The signal number of OUT output
	GD61042	The signal number of NET-IN input
	GD61043	The signal number of NET-OUT output
	GD61044	Temporary storage device of IN input function selection
	GD61046	Temporary storage device of OUT/NET-OUT output function selection
	GD61048	Temporary storage device of NET-IN input function selection
	GD61050 to GD61080	Backup device of NET-IN0 to NET-IN15
	GD61082 to GD61096	Backup device of IN0 to IN7
	GD61098 to GD61108	Backup device of OUT0 to OUT5
	GD61110	Backup device of the command position to be monitored
	GD61112	Backup device of the preset position
	GD61114	Backup device of the JOG travel amount (use the minimum travel amount of the GOT)
	GD61116	Type for waiting time of executing configuration
	GD61199	Station address backup
	GD61200+100n(n=0 to 30)	Display the operation number No. (Axis No.1 to Axis No.31)
	GD61202+100n(n=0 to 30)	NET-IN0 backup device (Axis No.1 to Axis No.31)
	GD61204+100n(n=0 to 30)	NET-IN1 backup device (Axis No.1 to Axis No.31)
	GD61206+100n(n=0 to 30)	NET-IN2 backup device (Axis No.1 to Axis No.31)
	GD61208+100n(n=0 to 30)	NET-IN3 backup device (Axis No.1 to Axis No.31)
	GD61210+100n(n=0 to 30)	NET-IN4 backup device (Axis No.1 to Axis No.31)
	GD61212+100n(n=0 to 30)	NET-IN5 backup device (Axis No.1 to Axis No.31)
	GD61214+100n(n=0 to 30)	NET-IN6 backup device (Axis No.1 to Axis No.31)
	GD61216+100n(n=0 to 30)	NET-IN7 backup device (Axis No.1 to Axis No.31)
	GD61218+100n(n=0 to 30)	NET-IN8 backup device (Axis No.1 to Axis No.31)
	GD61220+100n(n=0 to 30)	NET-IN9 backup device (Axis No.1 to Axis No.31)
	GD61222+100n(n=0 to 30)	NET-IN10 backup device (Axis No.1 to Axis No.31)
	GD61224+100n(n=0 to 30)	NET-IN11 backup device (Axis No.1 to Axis No.31)
	GD61226+100n(n=0 to 30)	NET-IN12 backup device (Axis No.1 to Axis No.31)
	GD61228+100n(n=0 to 30)	NET-IN13 backup device (Axis No.1 to Axis No.31)
	GD61230+100n(n=0 to 30)	NET-IN14 backup device (Axis No.1 to Axis No.31)
	GD61232+100n(n=0 to 30)	NET-IN15 backup device (Axis No.1 to Axis No.31)
	GD61234+100n(n=0 to 30)	IN0 backup device (Axis No.1 to Axis No.31)
	GD61236+100n(n=0 to 30)	IN1 backup device (Axis No.1 to Axis No.31)
	GD61238+100n(n=0 to 30)	IN2 backup device (Axis No.1 to Axis No.31)
	GD61240+100n(n=0 to 30)	IN3 backup device (Axis No.1 to Axis No.31)
	GD61242+100n(n=0 to 30)	IN4 backup device (Axis No.1 to Axis No.31)
	GD61244+100n(n=0 to 30)	IN5 backup device (Axis No.1 to Axis No.31)
	GD61246+100n(n=0 to 30)	IN6 backup device (Axis No.1 to Axis No.31)

Type	Device No.	Application
Word	GD61248+100n(n=0 to 30)	IN7 backup device (Axis No.1 to Axis No.31)
	GD61250+100n(n=0 to 30)	Backup device of the command position to be monitored (Axis No.1 to Axis No.31)
	GD61252+100n(n=0 to 30)	Backup device of the preset position (Axis No.1 to Axis No.31)
	GD61254+100n(n=0 to 30)	Backup device of the JOG travel amount (Axis No.1 to Axis No.31)
	GD61256+100n(n=0 to 30)	Type for waiting time of executing configuration (Axis No.1 to Axis No.31)
	GD64990 to GD64995	Clock digiswitch
	GD65000	Address number
	GD65001	Power supply type Specification
	GD65100 to GD65102	External control device (recipe)
	GD65103 to GD65105	External notification device (recipe)
	GS7	1 second binary counter
	GS513 to GS516	Change time
	GS531 to GS532	Monitor station disconnection (CH1)
	GS570 to GS576	MODBUS communication control function
	GS650 to GS652	Present time
	TMP950 to TMP996	For script operation

## 5.5 Comment List

Comment group No.	Comment No.	Where comments are used
497	No.1 to 240	B-31015、B-31018 to B-31019
498	No.1 to 60	B-31012、B-31013、B-31016、B-31017、B-31020、W-32004、W-32005
499	No.1 to 90	B-31012、B-31013、B-31016、B-31017、B-31020、W-32006
500	No.1、7	B-30002 to B-30005、B-30500 to B-31030
	No.2	B-30002
	No.3	B-30002、B-30003
	No.4	B-30002、B-30004
	No.5	B-30002、B-30005
	No.6	B-30002、B-30006
	No.8	B-30001
	No.9 to 13	B-30003
	No.14 to 18	B-30004
	No.19 to 20	B-30005、B-31014 to B-31019
	No.21 to 23	B-30005
	No.24 to 26	B-30006
	No.27	B-30003 to B-30006、B-31002、B-31010 to B-31014、B-31030
	No.28	B-30003 to B-30006、B-31002 to B-31008、B-31030
	No.29 to 31	B-30003 to B-30006、B-31002、B-31030
	No.32 to 33	B-31003 to B-31008
	No.34	B-31010 to B-31014
	No.35	B-31010 to B-31019
	No.36	B-30003 to B-30005、B-30500 to B-31019、B-31030
	No.37	B-30005、B-30500
	No.38 to 40	B-31015 to B-31019
	No.50 to 51	B-30001 to B-30006、B-31002 to B-31021、B-31030
	No.100 to 128	B-31002
	No.150 to 185	B-31004
	No.200 to 218	B-31005



Comment group No.	Comment No.	Where comments are used
500	No.250 to 269	B-31006
	No.300 to 316	B-31007
	No.350 to 362	B-31008
	No.400 to 413	B-31010
	No.450 to 457	B-31011
	No.500 to 524	B-31012
	No.550 to 582	B-31013
	No.600 to 602	B-31014
	No.650 to 660	B-31015
	No.700 to 702	B-31016
	No.750 to 752	B-31017
	No.800 to 806	B-31018
	No.850 to 856	B-31019
	No.900 to 902	B-31020
	No.950 to 961	B-31021
	No.1000 to 1014	B-31022 to B-31029
	No.1015 to 1018	B-31022
	No.1019 to 1022	B-31023
	No.1023 to 1026	B-31024
	No.1027 to 1030	B-31025
	No.1031 to 1034	B-31026
	No.1035 to 1038	B-31027
	No.1039 to 1042	B-31028
	No.1043 to 1045	B-31029
	No.1050 to 1064	B-31030
	No.1100 to 1101	W-30001
	No.1150	W-30002
	No.1200	W-30003
	No.1250 to 1284	W-30004
	No.1300 to 1334	W-32001
	No.1350 to 1354	W-32002
	No.1400 to 1403	W-32003
	No.1450 to 1451	W-32004 to W-32006
	No.1500 to 1503	W-32007
	No.1550 to 1552	W-32008

## 5.6 Script List

Item	Settings
Project script	Specified
Screen script	B-30500、B-31002、B-31021 to B-31029、W-30004、W-32001、W-32004 to W-32006
Object script	B-30500、W-30003

### 5.6.1 Project script

Script No.	30001	Script Name	Script30001
Comment	Initial Setting		
Data Type	Signed BIN16	Trigger Type	Rise, GB40
[w:GD60080]=201; //Set Document ID to 201 [w:GD60081]=1; //Set Document page No. to 1  [w:GD61000] = 1; // Initial value of Slave (driver) address number (number for Station No. switching)			
Script No.	31001	Script Name	Script31001
Comment	Ini value set for OpeDataScrnNo.		
Data Type	Signed BIN16	Trigger Type	Rise, GB40
// Initial value setting for the operation data screen No.  // Project script No.31001 // Start the script with a rise of GB40 (GOT always ON bit) // Execute this script only once after GOT power is ON.  [u16:GD61001] = 0; // Initial value of line 0 is No.0. [u16:GD61002] = 1; // Initial value of line 1 is No.1. [u16:GD61003] = 2; // Initial value of line 2 is No.2. [u16:GD61004] = 3; // Initial value of line 3 is No.3. [u16:GD61005] = 4; // Initial value of line 4 is No.4. [u16:GD61006] = 5; // Initial value of line 5 is No.5. [u16:GD61007] = 6; // Initial value of line 6 is No.6. [u16:GD61008] = 7; // Initial value of line 7 is No.7.			
Script No.	31002	Script Name	Script31002
Comment	Enable all comm with device		
Data Type	Signed BIN16	Trigger Type	Sampling, 1(Sec)
// Enable connection of all axis communication for connected device/equipment // Because multiple operation cannot use Station No. switching (Station No.= each CPU No. of driver), // it is necessary to disable connection of all axis communication. // Restoring from the multiple operation screen needs to enable the connection of all axis communication.  // Project script No.31002 // Start the script with 1 second cycle trigger  // Enable connection of all axis communication except multiple operation base screen if([w:GD60000] >= 30001 && [w:GD60000] <= 31021){ [w:GS531] = 0; // 0: Connect CH1station No.(driver address number) 0-15 axis 1: Disconnect [w:GS532] = 0; // 0: Connect of CH1station No.(driver address number) 16-31 axis 1: Disconnect }  // Enable all axis connection in the SYSTEM base screen if([w:GD60000] == 31030){ [w:GS531] = 0; [w:GS532] = 0; }			

Script No.	31014	Script Name	Script31014
Comment	Interlock		
Data Type	Signed BIN16	Trigger Type	Ordinary
<pre>// Script for interlock not to change screen // Turn interlock ON not to change the screen in case of despising overlap window, operation preparation, in operation  // Project script No.31014 // Start the script with always ON  // [b:GB62006] : Overlap window being displayed flag // [b:GB62007] : Interlock for screen switching // [b:GB62010] : Check before start/end operation script // [b:GB62013] : Preparing operation flag // [b:GB62015] : Touch panel in operation flag // [b:GB62016] : Operation-end start trigger  if([w:GD60004] != 0){ // In case overlap window is displaying     [b:GB62006] = 1; // Turns Overlap window being displayed flag ON } else { // In case overlap window is not displaying     [b:GB62006] = 0; // Clear the Overlap window being displayed flag }  // In case of displaying overlap window, check before start/end, operation preparation, touch panel in operation, operation-end in process // Turn Interlock ON not to change the screen [b:GB62007] = [b:GB62006]   [b:GB62010]   [b:GB62013]   [b:GB62015]   [b:GB62016];</pre>			

### 5.6.2 Screen script

#### Base screen 30500

Script No.	30002	Script name	Script30002
Comment	DocumentDisplayProcessOfLastPage		
Data type	Unsigned BIN16	Trigger type	Ordinary
<pre>//Check the total number of document pages is not 0. if([w:GD60082]!=0){     //Compare the current page number to the total number of document pages to see if the current page number exceeds the total number.     if([w:GD60081]&gt;[w:GD60082]){         //Set the last page to display.         [w:GD60081]=[w:GD60082];     } }</pre>			

#### Base screen 31002

Script No.	31003	Script Name	Script31003
Comment	Display upward		
Data Type	Signed BIN16	Trigger Type	Rise, GB62000
<pre>// Scrolls the displayed the operation data up/down // Calculate the number of upward counts of up/down scroll counter  // Operation data screen script No.31003 // Start the script with a rise of GB62000  [w:GD61010] = [w:GD61010] - 1; // Up/down scroll count value [w:GD61011] = [w:GD61011] - 16; // Modbus address offset value  if ([w:GD61010] &lt; 0) { [w:GD61010] = 0; } // Lock not to change when the upward button is touched while displaying No.0 if ([w:GD61011] &lt; 0) { [w:GD61011] = 0; } // Lock the Modbus offset in the same way  [b:GB62000] = OFF;</pre>			

Script No.	31004	Script Name	Script31004
Comment	Display downward		
Data Type	Signed BIN16	Trigger Type	Rise, GB62001
// Scrolls the displayed Operation data up/down // Calculate the number of downward counts of up/down scroll counter  // Operation data screen script No.31004 // Start the script with a rise of GB62001  [w:GD61010] = [w:GD61010] + 1;                    // Up/down scroll count value [w:GD61011] = [w:GD61011] + 16;                // Modbus address offset value  if ([w:GD61010] >= 8) { [w:GD61010] = 7; }    // Not display No.63 or higher after the Downward button is clicked 8 times or more. if ([w:GD61011] >= 128) { [w:GD61011] = 112; }    // Same as above (Modbus offset 16 multiplied by 7=112 No.56 is the starting point)  [b:GB62001] = OFF;			
Script No.	31013	Script Name	Script31013
Comment	Set position		
Data Type	Signed BIN16	Trigger Type	Rise, GB62019
// teaching set position // Script for teaching set position of the operation data screen  // Set position script No.31013 // Start the script with a rise of the Set position button (GB62019: bit momentary)  // [u16:GD61015] : Modbus offset value to display the command position while teaching  // Write the command position of the monitor where the operation data No. is selected // Adjust the command position by means of Min or Max If the command position exceed the specified limit if ([s32:400199] <= -8388608    [s32:400199] >= 8388607) { if ([s32:400199] <= -8388608) { [s32:401025[u16:GD61015]] = -8388608;    // Clip by means of Min } if ([s32:400199] >= 8388607) { [s32:401025[u16:GD61015]] = 8388607;    // Clip by means of Max } } else { [s32:401025[u16:GD61015]] = [s32:400199];    // With in the specified limit }  // Change the operation mode of the selected operation No. to Absolute (ABS) [s32:401281[u16:GD61015]] = 1;			
Script No.	31005	Script Name	Script31005
Comment	Read operation input data		
Data Type	Signed BIN16	Trigger Type	ON GB62002
// Read and display the value in the overlap window for the operation data input  // Operation data screen script No.31005 // [b:GB62002] : Starting trigger of this script using bit set  // [b:GB62002] : Starting trigger of this script using bit set // [w:GD61010] : Up/down scroll count value // [w:GD61011] : Modbus address offset value // [w:GD61012] : Touch position info of the operation data No. (which row) // [u16:GD61020] : Display operation number (No.)  [u16:GD61020] = [w:GD61012] + ([w:GD61010] * 8);    // Calculate the value of the operation No.  // Display the operation data No. according to the touched row switch ([w:GD61012])			

```

{
case 0 : [u16:GD61021] = [u16:401282[w:GD61011]]; // Operation mode of line 0
[s32:GD61022] = [s32:401025[w:GD61011]]; // Position of line 0
[u32:GD61024] = [u32:401153[w:GD61011]]; // Operating speed of line 0
[u16:GD61026] = [u16:401410[w:GD61011]]; // Operation function of line 0
[u32:GD61027] = [u32:401793[w:GD61011]]; // Push current of line 0
[u32:GD61029] = [u32:402049[w:GD61011]]; // Dwell time of line 0
[u16:GD61031] = [u16:401922[w:GD61011]]; // Sequential positioning of line 0
[u32:GD61032] = [u32:401537[w:GD61011]]; // Acceleration of line 0
[u32:GD61034] = [u32:401665[w:GD61011]]; // Deceleration of line 0
break;

case 1 : [u16:GD61021] = [u16:401284[w:GD61011]]; // Operation mode of line 1
[s32:GD61022] = [s32:401027[w:GD61011]]; // Position of line 1
[u32:GD61024] = [u32:401155[w:GD61011]]; // Operating speed of line 1
[u16:GD61026] = [u16:401412[w:GD61011]]; // Operation function of line 1
[u32:GD61027] = [u32:401795[w:GD61011]]; // Push current of line 1
[u32:GD61029] = [u32:402051[w:GD61011]]; // Dwell time of line 1
[u16:GD61031] = [u16:401924[w:GD61011]]; // Sequential positioning of line 1
[u32:GD61032] = [u32:401539[w:GD61011]]; // Acceleration of line 1
[u32:GD61034] = [u32:401667[w:GD61011]]; // Deceleration of line 1
break;

case 2 : [u16:GD61021] = [u16:401286[w:GD61011]]; // Operation mode of line 2
[s32:GD61022] = [s32:401029[w:GD61011]]; // Position of line 2
[u32:GD61024] = [u32:401157[w:GD61011]]; // Operating speed of line 2
[u16:GD61026] = [u16:401414[w:GD61011]]; // Operation function of line 2
[u32:GD61027] = [u32:401797[w:GD61011]]; // Push current of line 2
[u32:GD61029] = [u32:402053[w:GD61011]]; // Dwell time of line 2
[u16:GD61031] = [u16:401926[w:GD61011]]; // Sequential positioning of line 2
[u32:GD61032] = [u32:401541[w:GD61011]]; // Acceleration of line 2
[u32:GD61034] = [u32:401669[w:GD61011]]; // Deceleration of line 2
break;

case 3 : [u16:GD61021] = [u16:401288[w:GD61011]]; // Operation mode of line 3
[s32:GD61022] = [s32:401031[w:GD61011]]; // Position of line 3
[u32:GD61024] = [u32:401159[w:GD61011]]; // Operating speed of line 3
[u16:GD61026] = [u16:401416[w:GD61011]]; // Operation function of line 3
[u32:GD61027] = [u32:401799[w:GD61011]]; // Push current of line 3
[u32:GD61029] = [u32:402055[w:GD61011]]; // Dwell time of line 3
[u16:GD61031] = [u16:401928[w:GD61011]]; // Sequential positioning of line 3
[u32:GD61032] = [u32:401543[w:GD61011]]; // Acceleration of line 3
[u32:GD61034] = [u32:401671[w:GD61011]]; // Deceleration of line 3
break;

case 4 : [u16:GD61021] = [u16:401290[w:GD61011]]; // Operation mode of line 4
[s32:GD61022] = [s32:401033[w:GD61011]]; // Position of line 4
[u32:GD61024] = [u32:401161[w:GD61011]]; // Operating speed of line 4
[u16:GD61026] = [u16:401418[w:GD61011]]; // Operation function of line 4
[u32:GD61027] = [u32:401801[w:GD61011]]; // Push current of line 4
[u32:GD61029] = [u32:402057[w:GD61011]]; // Dwell time of line 4
[u16:GD61031] = [u16:401930[w:GD61011]]; // Sequential positioning of line 4
[u32:GD61032] = [u32:401545[w:GD61011]]; // Acceleration of line 4
[u32:GD61034] = [u32:401673[w:GD61011]]; // Deceleration of line 4
break;

case 5 : [u16:GD61021] = [u16:401292[w:GD61011]]; // Operation mode of line 5
[s32:GD61022] = [s32:401035[w:GD61011]]; // Position of line 5
[u32:GD61024] = [u32:401163[w:GD61011]]; // Operating speed of line 5
[u16:GD61026] = [u16:401420[w:GD61011]]; // Operation function of line 5
[u32:GD61027] = [u32:401803[w:GD61011]]; // Push current of line 5
[u32:GD61029] = [u32:402059[w:GD61011]]; // Dwell time of line 5
[u16:GD61031] = [u16:401932[w:GD61011]]; // Sequential positioning of line 5
[u32:GD61032] = [u32:401547[w:GD61011]]; // Acceleration of line 5

```

```

[u32:GD61034] = [u32:401675[w:GD61011]]; // Deceleration of line 5
break;

case 6 : [u16:GD61021] = [u16:401294[w:GD61011]]; // Operation mode of line 6
[s32:GD61022] = [s32:401037[w:GD61011]]; // Position of line 6
[u32:GD61024] = [u32:401165[w:GD61011]]; // Operating speed of line 6
[u16:GD61026] = [u16:401422[w:GD61011]]; // Operation function of line 6
[u32:GD61027] = [u32:401805[w:GD61011]]; // Push current of line 6
[u32:GD61029] = [u32:402061[w:GD61011]]; // Dwell time of line 6
[u16:GD61031] = [u16:401934[w:GD61011]]; // Sequential positioning of line 6
[u32:GD61032] = [u32:401549[w:GD61011]]; // Acceleration of line 6
[u32:GD61034] = [u32:401677[w:GD61011]]; // Deceleration of line 6
break;

case 7 : [u16:GD61021] = [u16:401296[w:GD61011]]; // Operation mode of line 7
[s32:GD61022] = [s32:401039[w:GD61011]]; // Position of line 7
[u32:GD61024] = [u32:401167[w:GD61011]]; // Operating speed of line 7
[u16:GD61026] = [u16:401424[w:GD61011]]; // Operation function of line 7
[u32:GD61027] = [u32:401807[w:GD61011]]; // Push current of line 7
[u32:GD61029] = [u32:402063[w:GD61011]]; // Dwell time of line 7
[u16:GD61031] = [u16:401936[w:GD61011]]; // Sequential positioning of line 7
[u32:GD61032] = [u32:401551[w:GD61011]]; // Acceleration of line 7
[u32:GD61034] = [u32:401679[w:GD61011]]; // Deceleration of line 7
break;
}

[b:GB62002] = 0; // End starting trigger of this script

```

#### Base screen 31002、31021

Script No.	31008	Script Name	Script31008
Comment	// DisplayCmdPositionWhenTestOpe		
Data Type	Signed BIN16	Trigger Type	Ordinary
// Display teaching of the operation data and command position of TEST while test operation			
// the operation data screen, Operation screen of TEST No.31008			
// Start the script with always ON			
// [u16:GD61020] : Display operation number (No.)			
[u16:GD61015] = [u16:GD61020] * 2; // Modbus offset to display command position while teaching			
Script No.	31011	Script Name	Script31011
Comment	In operation		
Data Type	Signed BIN16	Trigger Type	ON GB62015
// Script for positioning operation with touch panel operation			
// Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No.			
// After being changed, M0-M5 are turned ON by driver input command (400126).			
// In operation script No.31011			
// Start the script in case GB62015 in the operation preparation script (No.31010) is ON.			
// [b:GB62015] : Touch panel in operation flag (Starting trigger of this script)			
// Bit also works as interlock not to change the screen while touch panel in operation			
// [u16:400196] : Modbus address of selected data No.			
// [u16:GD61020] : Device for the operation data No.			
// In case of touch panel in operation			
if([b:GB62015] == 1){			
// Change M0-M5 of driver input command to reflect the selected operation data No.			
// To ease the touch panel processing load,			
// In case the currently selected data No. (400916) of the monitor and the selected operation data No. are not the same,			

```
// change the driver input command (400126) to reflect a change
```

```
if([u16:400196] != [u16:GD61020]){
    [b:400126.b0] = [b:GD61020.b0]; // M0
    [b:400126.b1] = [b:GD61020.b1]; // M1
    [b:400126.b2] = [b:GD61020.b2]; // M2
    [b:400126.b3] = [b:GD61020.b3]; // M3
    [b:400126.b4] = [b:GD61020.b4]; // M4
    [b:400126.b5] = [b:GD61020.b5]; // M5
}
```

### Base screen 31002, 31020, 31021

Script No.	31009	Script Name	Script31009
Comment	Check before start/end.		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62010
<pre>// Script to check whether an alarm is not occurring or Operation is not running before operation start/end // the operation data screen, I/O test for TEST, Screen script for operation (single operation)  // Check before start/end operation script No.31009 // Start the script with the [Start] button (2 seconds ON cycle) in each base screen  // [b:GB62010] : Starting trigger of this script using bit set // [b:GB62015] : Touch panel in operation flag  // Alarm check if([s32:400129] == 0){ // In case of no alarm     [b:GB62011] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)     [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking. } else { // in case the alarm occurring     [b:GB62011] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)     [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking. }  // In operation check if([s32:400201] == 0){ // Case: not in operation     [b:GB62012] = 1; // In operation checking flag (1: not in operation) } else { // In case the motor is running     [b:GB62012] = 0; // In operation checking flag (0: in operation)     [w:GD60004] = 32008; // Display the motor operating checking overlap window }  // Distinguish Operation start and Operation-end after finishing the check mentioned above if([b:GB62011] == 1 &amp;&amp; [b:GB62012] == 1){     if([b:GB62015] == 0){ // Prepare Operation start in case the touch panel is not in operation.         [b:GB62010] = 0; // End starting trigger of this script         [b:GB62013] = 1; // Operation preparation start trigger         [b:GB62011] = 0; // Clear checking flag to see if an alarm is occurring         [b:GB62012] = 0; // Clear the In operation checking flag     } else { // Prepare Operation-end in case the touch panel is in operation.         [b:GB62010] = 0; // End starting trigger of this script         [b:GB62016] = 1; // Operation-end start trigger         [b:GB62011] = 0; // Clear checking flag to see if an alarm is occurring         [b:GB62012] = 0; // Clear the In operation checking flag     } } else {     [b:GB62010] = 0; // End starting trigger of this script     [b:GB62011] = 0; // Clear checking flag to see if an alarm is occurring     [b:GB62012] = 0; // Clear the In operation checking flag }</pre>			
Script No.	31010	Script Name	Script31010
Comment	Operation preparation		
Data Type	Signed BIN16	Trigger Type	ON GB62013

```

// Script to prepare for touch panel operation
// Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter
// the operation data screen, I/O test for TEST, Screen script for operation (single operation)

// Operation preparation script No.31010
// Start the script in case GB62013 in the Check before start/end operation script (No.31009) is ON.

// [b:GB62013] : Operation preparation start trigger (start trigger of this script)
// [b:GB62014] : Preparing operation flag
// [b:GB62015] : Touch panel in operation flag
// [b:GB62018] : I/O test flag
// [u32:GD61050] : Backup device for NET-IN0-15
// [u32:GD61082] : Backup device for IN0-7
// [u32:GD61098] : Backup device for OUT0-5
// [s32:GD61110] : Backup device for Command position of a monitor
// [s32:GD61112] : Backup device for preset position
// [u32:GD61114] : Backup device for JOG moving distance (minimum moving distance for GOT)
// [w:GD61116] : Configuration execution waiting timer

// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62013] == 1 && [b:GB62014] == 0){

    [b:GB62014] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([u32:404449],[u32:GD61050],16); // Backs up NET-IN0-15
    bmov([u32:404353],[u32:GD61082],8); // Backs up IN0-7
    bmov([u32:404417],[u32:GD61098],6); // Backs up OUT0-5
    [s32:GD61112] = [s32:400909]; // Backs up the preset position
    [u32:GD61114] = [u32:404169]; // Backup JOG moving distance (used for Minimum JOG moving distance)

    // Clear the input value of NET-IN
    [w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [u32:404461] = 4; // NET-IN6 START
    [u32:404463] = 3; // NET-IN7 HOME
    [u32:404465] = 18; // NET-IN8 STOP
    [u32:404467] = 9; // NET-IN9 MS1
    [u32:404469] = 10; // NET-IN10 MS2
    [u32:404471] = 5; // NET-IN11 SSTART
    [u32:404473] = 6; // NET-IN12 +JOG
    [u32:404475] = 7; // NET-IN13 -JOG
    [u32:404477] = 1; // NET-IN14 FWD
    [u32:404479] = 2; // NET-IN15 RVS

    if([b:GB62018] == 0){ // In case of the operation data screen, Operation screen of TEST
        [u32:404449] = 48; // NET-IN0 M0
        [u32:404451] = 49; // NET-IN1 M1
        [u32:404453] = 50; // NET-IN2 M2
        [u32:404455] = 51; // NET-IN3 M3
        [u32:404457] = 52; // NET-IN4 M4
        [u32:404459] = 53; // NET-IN5 M5

        [u32:404353] = 32; // IN0 R0
        [u32:404355] = 33; // IN1 R1
        [u32:404357] = 34; // IN2 R2
        [u32:404359] = 35; // IN3 R3
        [u32:404361] = 36; // IN4 R4
        [u32:404363] = 37; // IN5 R5
        [u32:404365] = 18; // IN6 STOP
        [u32:404367] = 39; // IN7 R7

    } else { // In case of a TEST I/O test.

```



```

[u32:404449] = 40; // NET-IN0 R8
[u32:404451] = 41; // NET-IN1 R9
[u32:404453] = 42; // NET-IN2 R10
[u32:404455] = 43; // NET-IN3 R11
[u32:404457] = 44; // NET-IN4 R12
[u32:404459] = 45; // NET-IN5 R13

// Set IN/OUT to u16 in order to display the signal name in simulation
// In simulation, upper/lower reversal for 32 bit length cannot be done.
[u16:404354] = 32; // IN0 R0
[u16:404356] = 33; // IN1 R1
[u16:404358] = 34; // IN2 R2
[u16:404360] = 35; // IN3 R3
[u16:404362] = 36; // IN4 R4
[u16:404364] = 37; // IN5 R5
[u16:404366] = 38; // IN6 R6
[u16:404368] = 39; // IN7 R7

[u16:404418] = 40; // OUT0 R8
[u16:404420] = 41; // OUT1 R9
[u16:404422] = 42; // OUT2 R10
[u16:404424] = 43; // OUT3 R11
[u16:404426] = 44; // OUT4 R12
[u16:404428] = 45; // OUT5 R13
}

// Set the preset position parameter to the command position of the monitor
[s32:GD61110] = [s32:400199];
[s32:400909] = [s32:GD61110];

// Execution of Configuration
[s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61116] = [w:GS7];
}

```

```

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61116] > 1 && [b:GB62014] == 1){

```

```

[s32:400395] = 1; // Execute P-PRESET
[s32:400909] = [s32:GD61112]; // Restore the preset position parameter after backup
[s32:400397] = 0; // Zero clear execution of Configuration
[s32:400395] = 0; // Zero clear for the execution of P-PRESET

[b:GB62013] = 0; // Clear Operation preparation start trigger
// End the starting trigger in this script.
[b:GB62014] = 0; // Clear Preparing operation flag
[b:GB62015] = 1; // Touch panel in operation flag (1: in operation)
[b:GB62018] = 0; // Clear I/O test frag
}

```

Script No.	31012	Script Name	Script31012
Comment	Operation-end		
Data Type	Signed BIN16	Trigger Type	ON GB62016

```

// Script to disable touch panel operation
// Restore such as the NET-IN signal and the necessary parameter from backup
// the operation data screen, I/O test for TEST, Screen script for operation (single operation)

// Operation-end script No.31012
// Start the script in case GB62016 in the Check before start/end operation script (No.31009) is ON.

// [b:GB62015] : Touch panel in operation flag
// [b:GB62016] : Operation-end start trigger (start trigger of this script)
// [b:GB62017] : the operation-end in process flag

```

```

// [b:GB62018] : I/O test flag
// [u32:GD61050] : Backup device for NET-IN0-15
// [u32:GD61082] : Backup device for IN0-7
// [u32:GD61098] : Backup device for OUT0-5
// [s32:GD61110] : Backup device for Command position of a monitor
// [s32:GD61112] : Backup device for preset position
// [u32:GD61114] : Backup device for JOG moving distance (minimum moving distance for GOT)
// [w:GD61116] : Configuration execution waiting timer

// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62016] == 1 && [b:GB62017] == 0){

    [b:GB62015] = 0;           // Clear touch panel in operation flag
    [b:GB62017] = 1;           // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD61050],[u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD61082],[u32:404353],8); // Restore IN0-7 from backup
    bmov([u32:GD61098],[u32:404417],6); // Restore OUT0-5 from backup
    [u32:404169] = [u32:GD61114]; // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD61110] = [s32:400199];
    [s32:400909] = [s32:GD61110];

    // Clear the NET-IN input value when output test was conducted in the TEST I/O test screen
    [w:400126] = 0;

    // Execution of Configuration
    [s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD61116] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD61116] > 1 && [b:GB62017] == 1){

    [s32:400395] = 1;           // Executes P-PRESET
    [s32:400909] = [s32:GD61112]; // Reset the Preset position parameter
    [s32:400397] = 0;           // Zero clear execution of Configuration
    [s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62016] = 0;           // Clear Operation-end start trigger
                                // End the starting trigger in this script.
    [b:GB62017] = 0;           // Clear the operation-end in process flag
    [b:GB62018] = 0;           // Clear I/O test frag
}

```

#### Base screen 31022 to 31029

Script No.	31015	Script Name	Script31015
Comment	Multiple operation Interlock		
Data Type	Signed BIN16	Trigger Type	Ordinary
// Interlock is ON not to change the screen while multiple operation preparation/end, displaying overlap window // Screen script for 1-31 axis multiple operation  // Interlock script for multiple operation No.31015 // Start the script in case  // [b:GB62020] - [b:GB62320] : Starting trigger of each axis (1-31) using bit set // [b:GB62023] - [b:GB62323] : Operation preparation start trigger of each axis (1-31) // [b:GB62026] - [b:GB62326] : Operation-end start trigger of each axis (1-31) // [b:GB62006] : Overlap window being displayed flag // [b:GB62009] : Interlock flag for multiple operation preparation/end in process			

```

[b:GB62009] = [b:GB62006] | [b:GB62020] | [b:GB62023] | [b:GB62026] | [b:GB62030] | [b:GB62033] |
[b:GB62036];
[b:GB62009] = [b:GB62009] | [b:GB62040] | [b:GB62043] | [b:GB62046] | [b:GB62050] | [b:GB62053] |
[b:GB62056];
[b:GB62009] = [b:GB62009] | [b:GB62060] | [b:GB62063] | [b:GB62066] | [b:GB62070] | [b:GB62073] |
[b:GB62076];
[b:GB62009] = [b:GB62009] | [b:GB62080] | [b:GB62083] | [b:GB62086] | [b:GB62090] | [b:GB62093] |
[b:GB62096];
[b:GB62009] = [b:GB62009] | [b:GB62100] | [b:GB62103] | [b:GB62106] | [b:GB62110] | [b:GB62113] |
[b:GB62116];
[b:GB62009] = [b:GB62009] | [b:GB62120] | [b:GB62123] | [b:GB62126] | [b:GB62130] | [b:GB62133] |
[b:GB62136];
[b:GB62009] = [b:GB62009] | [b:GB62140] | [b:GB62143] | [b:GB62146] | [b:GB62150] | [b:GB62153] |
[b:GB62156];
[b:GB62009] = [b:GB62009] | [b:GB62160] | [b:GB62163] | [b:GB62166] | [b:GB62170] | [b:GB62173] |
[b:GB62176];
[b:GB62009] = [b:GB62009] | [b:GB62180] | [b:GB62183] | [b:GB62186] | [b:GB62190] | [b:GB62193] |
[b:GB62196];
[b:GB62009] = [b:GB62009] | [b:GB62200] | [b:GB62203] | [b:GB62206] | [b:GB62210] | [b:GB62213] |
[b:GB62216];
[b:GB62009] = [b:GB62009] | [b:GB62220] | [b:GB62223] | [b:GB62226] | [b:GB62230] | [b:GB62233] |
[b:GB62236];
[b:GB62009] = [b:GB62009] | [b:GB62240] | [b:GB62243] | [b:GB62246] | [b:GB62250] | [b:GB62253] |
[b:GB62256];
[b:GB62009] = [b:GB62009] | [b:GB62260] | [b:GB62263] | [b:GB62266] | [b:GB62270] | [b:GB62273] |
[b:GB62276];
[b:GB62009] = [b:GB62009] | [b:GB62280] | [b:GB62283] | [b:GB62286] | [b:GB62290] | [b:GB62293] |
[b:GB62296];
[b:GB62009] = [b:GB62009] | [b:GB62300] | [b:GB62303] | [b:GB62306] | [b:GB62310] | [b:GB62313] |
[b:GB62316];
[b:GB62009] = [b:GB62009] | [b:GB62320] | [b:GB62323] | [b:GB62326];

// Activate the interlock not to change other screen from the multiple operation screen
// [b:GB62025] - [b:GB62325] : In operation flag of each axis (1-31)
// [b:GB62008] : Multi operation interlock flag

[b:GB62008] = [b:GB62009] | [b:GB62025] | [b:GB62035] | [b:GB62045] | [b:GB62055] | [b:GB62065] |
[b:GB62075];
[b:GB62008] = [b:GB62008] | [b:GB62085] | [b:GB62095] | [b:GB62105] | [b:GB62115] | [b:GB62125] |
[b:GB62135];
[b:GB62008] = [b:GB62008] | [b:GB62145] | [b:GB62155] | [b:GB62165] | [b:GB62175] | [b:GB62185] |
[b:GB62195];
[b:GB62008] = [b:GB62008] | [b:GB62205] | [b:GB62215] | [b:GB62225] | [b:GB62235] | [b:GB62245] |
[b:GB62255];
[b:GB62008] = [b:GB62008] | [b:GB62265] | [b:GB62275] | [b:GB62285] | [b:GB62295] | [b:GB62305] |
[b:GB62315];
[b:GB62008] = [b:GB62008] | [b:GB62325];

// Clear multiple operation-end start flag in case of a destination error
// [w:GS281]、[w:GS282] :Device for station No. error

if ([b:GS281.b1] == 1) { [b:GB62020] = 0; }
if ([b:GS281.b2] == 1) { [b:GB62030] = 0; }
if ([b:GS281.b3] == 1) { [b:GB62040] = 0; }
if ([b:GS281.b4] == 1) { [b:GB62050] = 0; }
if ([b:GS281.b5] == 1) { [b:GB62060] = 0; }
if ([b:GS281.b6] == 1) { [b:GB62070] = 0; }
if ([b:GS281.b7] == 1) { [b:GB62080] = 0; }
if ([b:GS281.b8] == 1) { [b:GB62090] = 0; }
if ([b:GS281.b9] == 1) { [b:GB62100] = 0; }
if ([b:GS281.b10] == 1) { [b:GB62110] = 0; }
if ([b:GS281.b11] == 1) { [b:GB62120] = 0; }
if ([b:GS281.b12] == 1) { [b:GB62130] = 0; }
if ([b:GS281.b13] == 1) { [b:GB62140] = 0; }

```

```
if ([b:GS281.b14] == 1) { [b:GB62150] = 0; }  
if ([b:GS281.b15] == 1) { [b:GB62160] = 0; }  
  
if ([b:GS282.b0] == 1) { [b:GB62170] = 0; }  
if ([b:GS282.b1] == 1) { [b:GB62180] = 0; }  
if ([b:GS282.b2] == 1) { [b:GB62190] = 0; }  
if ([b:GS282.b3] == 1) { [b:GB62200] = 0; }  
if ([b:GS282.b4] == 1) { [b:GB62210] = 0; }  
if ([b:GS282.b5] == 1) { [b:GB62220] = 0; }  
if ([b:GS282.b6] == 1) { [b:GB62230] = 0; }  
if ([b:GS282.b7] == 1) { [b:GB62240] = 0; }  
if ([b:GS282.b8] == 1) { [b:GB62250] = 0; }  
if ([b:GS282.b9] == 1) { [b:GB62260] = 0; }  
if ([b:GS282.b10] == 1) { [b:GB62270] = 0; }  
if ([b:GS282.b11] == 1) { [b:GB62280] = 0; }  
if ([b:GS282.b12] == 1) { [b:GB62290] = 0; }  
if ([b:GS282.b13] == 1) { [b:GB62300] = 0; }  
if ([b:GS282.b14] == 1) { [b:GB62310] = 0; }  
if ([b:GS282.b15] == 1) { [b:GB62320] = 0; }
```

## Base screen 31022

Script No.	31016	Script Name	Script31016
Comment	1-4ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62020
<pre>// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end // Screen script for 1-4 axis multiple operation  // 1-4_Check before start/end multiple operation script No.31016 // Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of 1-4 axis  // [b:GB62020] : 1 axis starting trigger using bit set // [b:GB62030] : 2 axis starting trigger using bit set // [b:GB62040] : 3 axis starting trigger using bit set // [b:GB62050] : 4 axis starting trigger using bit set // [b:GB62025] : 1 axis touch panel in operation flag // [b:GB62035] : 2 axis touch panel in operation flag // [b:GB62045] : 3 axis touch panel in operation flag // [b:GB62055] : 4 axis touch panel in operation flag  // [w:GD61000] = 1 means as below // Because Occurring alarm checking screen is responsible for station No. switching // Station No.(CPU No. of driver) is needed for alarm clear (Therefore settings of each axis is required.) // Screen switching device (GD61199) is responsible for Station No. reset  // 1 axis if([b:GB62020] == 1){     [w:GD61000] = 1;      // Alarm check     if([1-1:s32:400129] == 0){        // In case of no alarm         [b:GB62021] = 1;            // Checking flag to see if an alarm is occurring(1: no alarm)         [w:GD60004] = 0;            // Clear the overlap window for occurring alarm checking.     } else {                        // in case the alarm occurring         [b:GB62021] = 0;            // Checking flag to see if an alarm is occurring(0: alarm occurring)         [w:GD60004] = 32007;        // Display the overlap window for occurring alarm checking.     }      // In operation check     if([1-1:s32:400201] == 0){        // Case: not in operation         [b:GB62022] = 1;            // In operation checking flag (1: not in operation)     } else {                        // In case the motor is running         [b:GB62022] = 0;            // In operation checking flag (0: in operation)         [w:GD60004] = 32008;        // Display the overlap window to check if a motor is in operation.     }      // Distinguish Operation start and Operation-end after finishing the check mentioned above     if([b:GB62021] == 1 &amp;&amp; [b:GB62022] == 1){         if([b:GB62025] == 0){        // Prepare Operation start in case the touch panel is not in operation.             [b:GB62020] = 0;        // End the starting trigger of this script             [b:GB62023] = 1;        // Operation preparation start trigger             [b:GB62021] = 0;        // Clear checking flag to see if an alarm is occurring             [b:GB62022] = 0;        // Clear the In operation checking flag         } else {                    // Prepare Operation-end in case the touch panel is in operation.             [b:GB62020] = 0;        // End the starting trigger of this script             [b:GB62026] = 1;        // Start trigger to terminate operation             [b:GB62021] = 0;        // Clear checking flag to see if an alarm is occurring             [b:GB62022] = 0;        // Clear the In operation checking flag         }     } else {         [b:GB62020] = 0;            // End the starting trigger of this script         [b:GB62021] = 0;            // Clear checking flag to see if an alarm is occurring     } }</pre>			

```

        [b:GB62022] = 0;    // Clear the In operation checking flag
    }
}

// 2 axis
if([b:GB62030] == 1){
    [w:GD61000] = 2;

    // Alarm check
    if([1-2:s32:400129] == 0){    // In case of no alarm
        [b:GB62031] = 1;    // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;    // Clear the overlap window for occurring alarm checking.
    } else {    // in case the alarm occurring
        [b:GB62031] = 0;    // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-2:s32:400201] == 0){    // Case: not in operation
        [b:GB62032] = 1;    // In operation checking flag (1: not in operation)
    } else {    // In case the motor is running
        [b:GB62032] = 0;    // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;    // Display the overlap window to check if a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62031] == 1 && [b:GB62032] == 1){
        if([b:GB62035] == 0){    // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62030] = 0;    // End the starting trigger of this script
            [b:GB62033] = 1;    // Operation preparation start trigger
            [b:GB62031] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62032] = 0;    // Clear the In operation checking flag
        } else {    // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62030] = 0;    // End the starting trigger of this script
            [b:GB62036] = 1;    // Start trigger to terminate operation
            [b:GB62031] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62032] = 0;    // Clear the In operation checking flag
        }
    } else {
        [b:GB62030] = 0;    // End the starting trigger of this script
        [b:GB62031] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62032] = 0;    // Clear the In operation checking flag
    }
}

// 3 axis
if([b:GB62040] == 1){
    [w:GD61000] = 3;

    // Alarm check
    if([1-3:s32:400129] == 0){    // In case of no alarm
        [b:GB62041] = 1;    // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;    // Clear the overlap window for occurring alarm checking.
    } else {    // in case the alarm occurring
        [b:GB62041] = 0;    // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-3:s32:400201] == 0){    // Case: not in operation
        [b:GB62042] = 1;    // In operation checking flag (1: not in operation)
    } else {    // In case the motor is running
        [b:GB62042] = 0;    // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;    // Display the overlap window to check if a motor is in operation.
    }
}

```

```

}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62041] == 1 && [b:GB62042] == 1){
    if([b:GB62045] == 0){ // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62040] = 0; // End the starting trigger of this script
        [b:GB62043] = 1; // Operation preparation start trigger
        [b:GB62041] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62042] = 0; // Clear the In operation checking flag
    } else { // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62040] = 0; // End the starting trigger of this script
        [b:GB62046] = 1; // Start trigger to terminate operation
        [b:GB62041] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62042] = 0; // Clear the In operation checking flag
    }
} else {
    [b:GB62040] = 0; // End the starting trigger of this script
    [b:GB62041] = 0; // Clear checking flag to see if an alarm is occurring
    [b:GB62042] = 0; // Clear the In operation checking flag
}
}

// 4 axis
if([b:GB62050] == 1){
    [w:GD61000] = 4;

    // Alarm check
    if([1-4:s32:400129] == 0){ // In case of no alarm
        [b:GB62051] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62051] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }
}

// In operation check
if([1-4:s32:400201] == 0){ // Case: not in operation
    [b:GB62052] = 1; // In operation checking flag (1: not in operation)
} else { // In case the motor is running
    [b:GB62052] = 0; // In operation checking flag (0: in operation)
    [w:GD60004] = 32008; // Display the overlap window to check if a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62051] == 1 && [b:GB62052] == 1){
    if([b:GB62055] == 0){ // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62050] = 0; // End the starting trigger of this script
        [b:GB62053] = 1; // Operation preparation start trigger
        [b:GB62051] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62052] = 0; // Clear the In operation checking flag
    } else { // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62050] = 0; // End the starting trigger of this script
        [b:GB62056] = 1; // Start trigger to terminate operation
        [b:GB62051] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62052] = 0; // Clear the In operation checking flag
    }
} else {
    [b:GB62050] = 0; // End the starting trigger of this script
    [b:GB62051] = 0; // Clear checking flag to see if an alarm is occurring
    [b:GB62052] = 0; // Clear the In operation checking flag
}
}

```

Script No.	31017	Script Name	Script31017
Comment	1-4_Multi ope prep		

Data Type	Signed BIN16	Trigger Type	ON GB62023
<pre> // Script to prepare for touch panel operation // Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter // Screen script for 1-4 axis multiple operation  // 1-4_Multiple operation preparation script No.31017 // Start the script in case GB62023, GB62033, GB62043, GB62053 in the 1-4_Check before start/end multiple operation script (No.31016) are ON.  // [b:GB62023], [b:GB62033], [b:GB62043], [b:GB62053] : Operation preparation start trigger (start trigger of this script) // [b:GB62024], [b:GB62034], [b:GB62044], [b:GB62054] : Preparing operation flag // [b:GB62025], [b:GB62035], [b:GB62045], [b:GB62055] : Touch panel in operation flag // [u32:GD61202],[u32:GD61302],[u32:GD61402],[u32:GD61502] : Backup device for NET-IN0-15 // [u32:GD61234],[u32:GD61334],[u32:GD61434],[u32:GD61534] : Backup device for IN0-7 // [s32:GD61250], [s32:GD61350], [s32:GD61450], [s32:GD61550] : Backup device for Command position of a monitor // [s32:GD61252], [s32:GD61352], [s32:GD61452], [s32:GD61552] : Backup device for preset position // [u32:GD61254],[u32:GD61354],[u32:GD61454],[u32:GD61554] : Backup device for JOG moving distance (minimum moving distance for GOT) // [w:GD61256], [w:GD61356], [w:GD61456], [w:GD61556] : Configuration execution waiting timer  // 1 axis // In case Operation-preparation start trigger and Operation-preparation are not completed. if([b:GB62023] == 1 &amp;&amp; [b:GB62024] == 0){      [b:GB62024] = 1;                // Preparing operation flag (1: Preparing operation)     bmov([1-1:u32:404449],[u32:GD61202],16); // Backs up NET-IN0-15     bmov([1-1:u32:404353],[u32:GD61234],8);  // Backs up IN0-7     [s32:GD61252] = [1-1:s32:400909];        // Backs up the preset position     [u32:GD61254] = [1-1:u32:404169];        // Backup JOG moving distance (used for Minimum JOG moving distance)      // Clear the input value of NET-IN     [1-1:w:400126] = 0;      // Change the signal assignment for touch panel operation     [1-1:u32:404449] = 48; // NET-IN0  M0     [1-1:u32:404451] = 49; // NET-IN1  M1     [1-1:u32:404453] = 50; // NET-IN2  M2     [1-1:u32:404455] = 51; // NET-IN3  M3     [1-1:u32:404457] = 52; // NET-IN4  M4     [1-1:u32:404459] = 53; // NET-IN5  M5     [1-1:u32:404461] = 4;  // NET-IN6  START     [1-1:u32:404463] = 3;  // NET-IN7  HOME     [1-1:u32:404465] = 18; // NET-IN8  STOP     [1-1:u32:404467] = 9;  // NET-IN9  MS1     [1-1:u32:404469] = 10; // NET-IN10 MS2     [1-1:u32:404471] = 5;  // NET-IN11 SSTART     [1-1:u32:404473] = 6;  // NET-IN12 +JOG     [1-1:u32:404475] = 7;  // NET-IN13 -JOG     [1-1:u32:404477] = 1;  // NET-IN14 FWD     [1-1:u32:404479] = 2;  // NET-IN15 RVS      [1-1:u32:404353] = 32; // IN0  R0     [1-1:u32:404355] = 33; // IN1  R1     [1-1:u32:404357] = 34; // IN2  R2     [1-1:u32:404359] = 35; // IN3  R3     [1-1:u32:404361] = 36; // IN4  R4     [1-1:u32:404363] = 37; // IN5  R5     [1-1:u32:404365] = 18; // IN6  STOP     [1-1:u32:404367] = 39; // IN7  R7      // Set the preset position parameter to the command position of the monitor </pre>			



```

[s32:GD61250] = [1-1:s32:400199];
[1-1:s32:400909] = [s32:GD61250];

// Execution of Configuration
[1-1:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61256] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61256] > 1 && [b:GB62024] == 1){

    [1-1:s32:400395] = 1;           // Executes P-PRESET
    [1-1:s32:400909] = [s32:GD61252]; // Restore the preset position parameter after backup
    [1-1:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-1:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62023] = 0;               // Clear Operation preparation start trigger
                                   // End the starting trigger in this script.
    [b:GB62024] = 0;               // Clear Preparing operation flag
    [b:GB62025] = 1;               // Touch panel in operation flag (1: in operation)
}

// 2 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62033] == 1 && [b:GB62034] == 0){

    [b:GB62034] = 1;               // Preparing operation flag (1: Preparing operation)
    bmov([1-2:u32:404449],[u32:GD61302],16); // Backs up NET-IN0-15
    bmov([1-2:u32:404353],[u32:GD61334],8);  // Backs up IN0-7
    [s32:GD61352] = [1-2:s32:400909];        // Backs up the preset position
    [u32:GD61354] = [1-2:u32:404169];        // Backup JOG moving distance (used for Minimum JOG moving
distance)

    // Clear the input value of NET-IN
    [1-2:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-2:u32:404449] = 48; // NET-IN0  M0
    [1-2:u32:404451] = 49; // NET-IN1  M1
    [1-2:u32:404453] = 50; // NET-IN2  M2
    [1-2:u32:404455] = 51; // NET-IN3  M3
    [1-2:u32:404457] = 52; // NET-IN4  M4
    [1-2:u32:404459] = 53; // NET-IN5  M5
    [1-2:u32:404461] = 4;  // NET-IN6  START
    [1-2:u32:404463] = 3;  // NET-IN7  HOME
    [1-2:u32:404465] = 18; // NET-IN8  STOP
    [1-2:u32:404467] = 9;  // NET-IN9  MS1
    [1-2:u32:404469] = 10; // NET-IN10 MS2
    [1-2:u32:404471] = 5;  // NET-IN11 SSTART
    [1-2:u32:404473] = 6;  // NET-IN12 +JOG
    [1-2:u32:404475] = 7;  // NET-IN13 -JOG
    [1-2:u32:404477] = 1;  // NET-IN14 FWD
    [1-2:u32:404479] = 2;  // NET-IN15 RVS

    [1-2:u32:404353] = 32; // IN0 R0
    [1-2:u32:404355] = 33; // IN1 R1
    [1-2:u32:404357] = 34; // IN2 R2
    [1-2:u32:404359] = 35; // IN3 R3
    [1-2:u32:404361] = 36; // IN4 R4
    [1-2:u32:404363] = 37; // IN5 R5
    [1-2:u32:404365] = 18; // IN6 STOP
    [1-2:u32:404367] = 39; // IN7 R7

```

```

// Set the preset position parameter to the command position of the monitor
[s32:GD61350] = [1-2:s32:400199];
[1-2:s32:400909] = [s32:GD61350];

// Execution of Configuration
[1-2:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61356] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61356] > 1 && [b:GB62034] == 1){

    [1-2:s32:400395] = 1;           // Executes P-PRESET
    [1-2:s32:400909] = [s32:GD61352]; // Restore the preset position parameter after backup
    [1-2:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-2:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62033] = 0;               // Clear Operation preparation start trigger
                                   // End the starting trigger in this script.
    [b:GB62034] = 0;               // Clear Preparing operation flag
    [b:GB62035] = 1;               // Touch panel in operation flag (1: in operation)
}

// 3 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62043] == 1 && [b:GB62044] == 0){

    [b:GB62044] = 1;               // Preparing operation flag (1: Preparing operation)
    bmov([1-3:u32:404449],[u32:GD61402],16); // Backs up NET-IN0-15
    bmov([1-3:u32:404353],[u32:GD61434],8);  // Backs up IN0-7
    [s32:GD61452] = [1-3:s32:400909];        // Backs up the preset position
    [u32:GD61454] = [1-3:u32:404169];        // Backup JOG moving distance (used for Minimum JOG moving
distance)

    // Clear the input value of NET-IN
    [1-3:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-3:u32:404449] = 48; // NET-IN0  M0
    [1-3:u32:404451] = 49; // NET-IN1  M1
    [1-3:u32:404453] = 50; // NET-IN2  M2
    [1-3:u32:404455] = 51; // NET-IN3  M3
    [1-3:u32:404457] = 52; // NET-IN4  M4
    [1-3:u32:404459] = 53; // NET-IN5  M5
    [1-3:u32:404461] = 4;  // NET-IN6  START
    [1-3:u32:404463] = 3;  // NET-IN7  HOME
    [1-3:u32:404465] = 18; // NET-IN8  STOP
    [1-3:u32:404467] = 9;  // NET-IN9  MS1
    [1-3:u32:404469] = 10; // NET-IN10 MS2
    [1-3:u32:404471] = 5;  // NET-IN11 SSTART
    [1-3:u32:404473] = 6;  // NET-IN12 +JOG
    [1-3:u32:404475] = 7;  // NET-IN13 -JOG
    [1-3:u32:404477] = 1;  // NET-IN14 FWD
    [1-3:u32:404479] = 2;  // NET-IN15 RVS

    [1-3:u32:404353] = 32; // IN0 R0
    [1-3:u32:404355] = 33; // IN1 R1
    [1-3:u32:404357] = 34; // IN2 R2
    [1-3:u32:404359] = 35; // IN3 R3
    [1-3:u32:404361] = 36; // IN4 R4
    [1-3:u32:404363] = 37; // IN5 R5

```

```

[1-3:u32:404365] = 18; // IN6 STOP
[1-3:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD61450] = [1-3:s32:400199];
[1-3:s32:400909] = [s32:GD61450];

// Execution of Configuration
[1-3:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61456] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61456] > 1 && [b:GB62044] == 1){

    [1-3:s32:400395] = 1; // Executes P-PRESET
    [1-3:s32:400909] = [s32:GD61452]; // Restore the preset position parameter after backup
    [1-3:s32:400397] = 0; // Zero clear execution of Configuration
    [1-3:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62043] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62044] = 0; // Clear Preparing operation flag
    [b:GB62045] = 1; // Touch panel in operation flag (1: in operation)
}

// 4 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62053] == 1 && [b:GB62054] == 0){

    [b:GB62054] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-4:u32:404449],[u32:GD61502],16); // Backs up NET-IN0-15
    bmov([1-4:u32:404353],[u32:GD61534],8); // Backs up IN0-7
    [s32:GD61552] = [1-4:s32:400909]; // Backs up the preset position
    [u32:GD61554] = [1-4:u32:404169]; // Backup JOG moving distance (used for Minimum JOG moving
distance)

    // Clear the input value of NET-IN
    [1-4:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-4:u32:404449] = 48; // NET-IN0 M0
    [1-4:u32:404451] = 49; // NET-IN1 M1
    [1-4:u32:404453] = 50; // NET-IN2 M2
    [1-4:u32:404455] = 51; // NET-IN3 M3
    [1-4:u32:404457] = 52; // NET-IN4 M4
    [1-4:u32:404459] = 53; // NET-IN5 M5
    [1-4:u32:404461] = 4; // NET-IN6 START
    [1-4:u32:404463] = 3; // NET-IN7 HOME
    [1-4:u32:404465] = 18; // NET-IN8 STOP
    [1-4:u32:404467] = 9; // NET-IN9 MS1
    [1-4:u32:404469] = 10; // NET-IN10 MS2
    [1-4:u32:404471] = 5; // NET-IN11 SSTART
    [1-4:u32:404473] = 6; // NET-IN12 +JOG
    [1-4:u32:404475] = 7; // NET-IN13 -JOG
    [1-4:u32:404477] = 1; // NET-IN14 FWD
    [1-4:u32:404479] = 2; // NET-IN15 RVS

    [1-4:u32:404353] = 32; // IN0 R0
    [1-4:u32:404355] = 33; // IN1 R1
    [1-4:u32:404357] = 34; // IN2 R2
    [1-4:u32:404359] = 35; // IN3 R3

```

```

[1-4:u32:404361] = 36; // IN4 R4
[1-4:u32:404363] = 37; // IN5 R5
[1-4:u32:404365] = 18; // IN6 STOP
[1-4:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD61550] = [1-4:s32:400199];
[1-4:s32:400909] = [s32:GD61550];

// Execution of Configuration
[1-4:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61556] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61556] > 1 && [b:GB62054] == 1){

    [1-4:s32:400395] = 1; // Executes P-PRESET
    [1-4:s32:400909] = [s32:GD61552]; // Restore the preset position parameter after backup
    [1-4:s32:400397] = 0; // Zero clear execution of Configuration
    [1-4:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62053] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62054] = 0; // Clear Preparing operation flag
    [b:GB62055] = 1; // Touch panel in operation flag (1: in operation)
}

```

Script No.	31018	Script Name	Script31018
Comment	1-4_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62025

```

// Script for positioning operation with touch panel operation
// Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No.
// After being changed, M0-M5 are turned ON by driver input command (400126).

// 1-4_Script in multiple operation No.31018
// Start the script in case GB62025, GB62035, GB62045, GB62055 in the 1-4_multiple operation preparation
script (No.31017) are ON.

// [b:GB62025],[b:GB62035],[b:GB62045],[b:GB62055] : Touch panel in operation flag(start trigger of this
script)
// Bit also works as interlock not to change the screen while touch panel in operation

// [1-1:u32:400195] - [1-4:u32:400195] : Modbus address of selected data No.
// [u32:GD61200], [u32GD61300], [GD61400], [u32:GD61500] : Device for the operation data No.

// Change M0-M5 of driver input command to reflect the selected operation data No.
// To ease the touch panel processing load,
// In case the currently selected data No. (400915) of the monitor and the selected operation data No. are not
the same,
// change the driver input command (400126) to reflect a change

// 1 axis
if([b:GB62025] == 1){

    if([1-1:u32:400195] != [u32:GD61200]){
        [1-1:b:400126.b0] = [b:GD61200.b0]; // M0
        [1-1:b:400126.b1] = [b:GD61200.b1]; // M1
        [1-1:b:400126.b2] = [b:GD61200.b2]; // M2
        [1-1:b:400126.b3] = [b:GD61200.b3]; // M3
        [1-1:b:400126.b4] = [b:GD61200.b4]; // M4
        [1-1:b:400126.b5] = [b:GD61200.b5]; // M5
    }
}

```

```

}

// 2 axis
if([b:GB62035] == 1){

    if([1-2:u32:400195] != [u32:GD61300]){
        [1-2:b:400126.b0] = [b:GD61300.b0];    // M0
        [1-2:b:400126.b1] = [b:GD61300.b1];    // M1
        [1-2:b:400126.b2] = [b:GD61300.b2];    // M2
        [1-2:b:400126.b3] = [b:GD61300.b3];    // M3
        [1-2:b:400126.b4] = [b:GD61300.b4];    // M4
        [1-2:b:400126.b5] = [b:GD61300.b5];    // M5
    }
}

// 3 axis
if([b:GB62045] == 1){

    if([1-3:u32:400195] != [u32:GD61400]){
        [1-3:b:400126.b0] = [b:GD61400.b0];    // M0
        [1-3:b:400126.b1] = [b:GD61400.b1];    // M1
        [1-3:b:400126.b2] = [b:GD61400.b2];    // M2
        [1-3:b:400126.b3] = [b:GD61400.b3];    // M3
        [1-3:b:400126.b4] = [b:GD61400.b4];    // M4
        [1-3:b:400126.b5] = [b:GD61400.b5];    // M5
    }
}

// 4 axis
if([b:GB62055] == 1){

    if([1-4:u32:400195] != [u32:GD61500]){
        [1-4:b:400126.b0] = [b:GD61500.b0];    // M0
        [1-4:b:400126.b1] = [b:GD61500.b1];    // M1
        [1-4:b:400126.b2] = [b:GD61500.b2];    // M2
        [1-4:b:400126.b3] = [b:GD61500.b3];    // M3
        [1-4:b:400126.b4] = [b:GD61500.b4];    // M4
        [1-4:b:400126.b5] = [b:GD61500.b5];    // M5
    }
}
}

```

Script No.	31019	Script Name	Script31019
Comment	1-4_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62026

```

// Script to disable touch panel operation
// Restore such as the NET-IN signal and the necessary parameter from backup
// Screen script for 1-4 axis multiple operation

// 1-4_End multiple operation script No.31019
// Start the script in case GB62026, GB62036, GB62046, GB62056 in the 1-4_Check before start/end multiple
operation script (No.31016) are ON.

// [b:GB62025], [b:GB62035], [b:GB62045], [b:GB62055] : Touch panel in operation flag
// [b:GB62026], [b:GB62036], [b:GB62046], [b:GB62056] : Operation-end start trigger (start trigger of this
script)
// [b:GB62027], [b:GB62037], [b:GB62047], [b:GB62057] : the operation-end in process flag
// [u32:GD61202], [u32:GD61302], [u32:GD61402], [u32:GD61502] : Backup device for NET-IN0-15
// [u32:GD61234], [u32:GD61334], [u32:GD61434], [u32:GD61534] : Backup device for IN0-7
// [s32:GD61250], [s32:GD61350], [s32:GD61450], [s32:GD61550] : Backup device for Command position of
a monitor
// [s32:GD61252], [s32:GD61352], [s32:GD61452], [s32:GD61552] : Backup device for preset position
// [u32:GD61254],[u32:GD61354],[u32:GD61454],[u32:GD61554] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD61256], [w:GD61356], [w:GD61456], [w:GD61556] : Configuration execution waiting timer

```

```

// 1 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62026] == 1 && [b:GB62027] == 0){

    [b:GB62025] = 0;                // Clear touch panel in operation flag
    [b:GB62027] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD61202],[1-1:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD61234],[1-1:u32:404353],8);  // Restore IN0-7 from backup
    [1-1:u32:404169] = [u32:GD61254];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD61250] = [1-1:s32:400199];
    [1-1:s32:400909] = [s32:GD61250];

    // Clear the input value of NET-IN
    [1-1:w:400126] = 0;

    // Execution of Configuration
    [1-1:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD61256] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD61256] > 1 && [b:GB62027] == 1){

    [1-1:s32:400395] = 1;          // Executes P-PRESET
    [1-1:s32:400909] = [s32:GD61252]; // Reset the Preset position parameter
    [1-1:s32:400397] = 0;          // Zero clear execution of Configuration
    [1-1:s32:400395] = 0;          // Zero clear for the execution of P-PRESET

    [b:GB62026] = 0;              // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62027] = 0;              // Clear the operation-end in process flag
}

// 2 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62036] == 1 && [b:GB62037] == 0){

    [b:GB62035] = 0;                // Clear touch panel in operation flag
    [b:GB62037] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD61302],[1-2:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD61334],[1-2:u32:404353],8);  // Restore IN0-7 from backup
    [1-2:u32:404169] = [u32:GD61354];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD61350] = [1-2:s32:400199];
    [1-2:s32:400909] = [s32:GD61350];

    // Clear the input value of NET-IN
    [1-2:w:400126] = 0;

    // Execution of Configuration
    [1-2:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD61356] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD61356] > 1 && [b:GB62037] == 1){

```

```

[1-2:s32:400395] = 1;           // Executes P-PRESET
[1-2:s32:400909] = [s32:GD61352]; // Reset the Preset position parameter
[1-2:s32:400397] = 0;           // Zero clear execution of Configuration
[1-2:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

[b:GB62036] = 0;               // Clear Operation-end start trigger
                                // End the starting trigger in this script.
[b:GB62037] = 0;               // Clear the operation-end in process flag
}

// 3 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62046] == 1 && [b:GB62047] == 0){

    [b:GB62045] = 0;             // Clear touch panel in operation flag
    [b:GB62047] = 1;             // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD61402],[1-3:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD61434],[1-3:u32:404353],8);  // Restore IN0-7 from backup
    [1-3:u32:404169] = [u32:GD61454];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD61450] = [1-3:s32:400199];
    [1-3:s32:400909] = [s32:GD61450];

    // Clear the input value of NET-IN
    [1-3:w:400126] = 0;

    // Execution of Configuration
    [1-3:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD61456] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD61456] > 1 && [b:GB62047] == 1){

    [1-3:s32:400395] = 1;           // Executes P-PRESET
    [1-3:s32:400909] = [s32:GD61452]; // Reset the Preset position parameter
    [1-3:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-3:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62046] = 0;               // Clear Operation-end start trigger
                                // End the starting trigger in this script.
    [b:GB62047] = 0;               // Clear the operation-end in process flag
}

// 4 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62056] == 1 && [b:GB62057] == 0){

    [b:GB62055] = 0;             // Clear touch panel in operation flag
    [b:GB62057] = 1;             // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD61502],[1-4:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD61534],[1-4:u32:404353],8);  // Restore IN0-7 from backup
    [1-4:u32:404169] = [u32:GD61554];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD61550] = [1-4:s32:400199];
    [1-4:s32:400909] = [s32:GD61550];

    // Clear the input value of NET-IN
    [1-4:w:400126] = 0;

```

```

// Execution of Configuration
[1-4:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61556] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD61556] > 1 && [b:GB62057] == 1){

    [1-4:s32:400395] = 1;           // Executes P-PRESET
    [1-4:s32:400909] = [s32:GD61552]; // Reset the Preset position parameter
    [1-4:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-4:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62056] = 0;               // Clear Operation-end start trigger
                                // End the starting trigger in this script.
    [b:GB62057] = 0;               // Clear the operation-end in process flag
}

```

Script No.	31016	Script Name	Script31016
Comment	1-4ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) B62030
Same as the script No. 31016 for the base screen 31022			
Script No.	31017	Script Name	Script31017
Comment	1-4_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62033
Same as the script No. 31017 for the base screen 31022			
Script No.	31018	Script Name	Script31018
Comment	1-4_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62035
Same as the script No. 31018 for the base screen 31022			
Script No.	31019	Script Name	Script31019
Comment	1-4_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62036
Same as the script No. 31019 for the base screen 31022			
Script No.	31016	Script Name	Script31016
Comment	1-4ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62040
Same as the script No. 31016 for the base screen 31022			
Script No.	31017	Script Name	Script31017
Comment	1-4_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62043
Same as the script No. 31017 for the base screen 31022			
Script No.	31018	Script Name	Script31018
Comment	1-4_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62045
Same as the script No. 31018 for the base screen 31022			
Script No.	31019	Script Name	Script31019
Comment	1-4_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62046



Same as the script No. 31019 for the base screen 31022			
Script No.	31016	Script Name	Script31016
Comment	1-4ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62050
Same as the script No. 31016 for the base screen 31022			
Script No.	31017	Script Name	Script31017
Comment	1-4_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62053
Same as the script No. 31017 for the base screen 31022			
Script No.	31018	Script Name	Script31018
Comment	1-4_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62055
Same as the script No. 31018 for the base screen 31022			
Script No.	31019	Script Name	Script31019
Comment	1-4_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62056
Same as the script No. 31019 for the base screen 31022			

#### Base screen 31023

Script No.	31020	Script Name	Script31020
Comment	5-8ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62060
// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end // Screen script for 5-8 axis multiple operation  // 5-8_Check before start/end multiple operation script No.31020 // Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of 5-8 axis  // [b:GB62060] : 5 axis starting trigger using bit set // [b:GB62070] : 6 axis starting trigger using bit set // [b:GB62080] : 7 axis starting trigger using bit set // [b:GB62090] : 8 axis starting trigger using bit set // [b:GB62065] : 5 axis touch panel in operation flag // [b:GB62075] : 6 axis touch panel in operation flag // [b:GB62085] : 7 axis touch panel in operation flag // [b:GB62095] : 8 axis touch panel in operation flag  // [w:GD61000] = 1 means as below // Because Occurring alarm checking screen is responsible for station No. switching // Station No.(CPU No. of driver) is needed for alarm clear (Therefore settings of each axis is required.) // Screen switching device (GD61199) is responsible for Station No. reset  // 5 axis if([b:GB62060] == 1){ [w:GD61000] = 5;  // Alarm check if([1-5:s32:400129] == 0){ // In case of no alarm [b:GB62061] = 1; // Checking flag to see if an alarm is occurring(1: no alarm) [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking. } else { // in case the alarm occurring [b:GB62061] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)			

```

    [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
}

// In operation check
if([1-5:s32:400201] == 0){    // Case: not in operation
    [b:GB62062] = 1;        // In operation checking flag (1: not in operation)
} else {                    // In case the motor is running
    [b:GB62062] = 0;        // In operation checking flag (0: in operation)
    [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62061] == 1 && [b:GB62062] == 1){
    if([b:GB62065] == 0){    // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62060] = 0;    // End the starting trigger of this script
        [b:GB62063] = 1;    // Operation preparation start trigger
        [b:GB62061] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62062] = 0;    // Clear the In operation checking flag
    } else {                // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62060] = 0;    // End the starting trigger of this script
        [b:GB62066] = 1;    // Start trigger to terminate operation
        [b:GB62061] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62062] = 0;    // Clear the In operation checking flag
    }
} else {
    [b:GB62060] = 0;        // End the starting trigger of this script
    [b:GB62061] = 0;        // Clear checking flag to see if an alarm is occurring
    [b:GB62062] = 0;        // Clear the In operation checking flag
}
}

// 6 axis
if([b:GB62070] == 1){
    [w:GD61000] = 6;

    // Alarm check
    if([1-6:s32:400129] == 0){    // In case of no alarm
        [b:GB62071] = 1;        // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;        // Clear the overlap window for occurring alarm checking.
    } else {                    // in case the alarm occurring
        [b:GB62071] = 0;        // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-6:s32:400201] == 0){    // Case: not in operation
        [b:GB62072] = 1;        // In operation checking flag (1: not in operation)
    } else {                    // In case the motor is running
        [b:GB62072] = 0;        // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62071] == 1 && [b:GB62072] == 1){
        if([b:GB62075] == 0){    // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62070] = 0;    // End the starting trigger of this script
            [b:GB62073] = 1;    // Operation preparation start trigger
            [b:GB62071] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62072] = 0;    // Clear the In operation checking flag
        } else {                // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62070] = 0;    // End the starting trigger of this script
            [b:GB62076] = 1;    // Start trigger to terminate operation
            [b:GB62071] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62072] = 0;    // Clear the In operation checking flag
        }
    }
}

```

```

    }
  } else {
    [b:GB62070] = 0;      // End the starting trigger of this script
    [b:GB62071] = 0;      // Clear checking flag to see if an alarm is occurring
    [b:GB62072] = 0;      // Clear the In operation checking flag
  }
}

// 7 axis
if([b:GB62080] == 1){
  [w:GD61000] = 7;

  // Alarm check
  if([1-7:s32:400129] == 0){      // In case of no alarm
    [b:GB62081] = 1;              // Checking flag to see if an alarm is occurring(1: no alarm)
    [w:GD60004] = 0;              // Clear the overlap window for occurring alarm checking.
  } else {                        // in case the alarm occurring
    [b:GB62081] = 0;              // Checking flag to see if an alarm is occurring(0: alarm occurring)
    [w:GD60004] = 32007;          // Display the overlap window for occurring alarm checking.
  }

  // In operation check
  if([1-7:s32:400201] == 0){      // Case: not in operation
    [b:GB62082] = 1;              // In operation checking flag (1: not in operation)
  } else {                        // In case the motor is running
    [b:GB62082] = 0;              // In operation checking flag (0: in operation)
    [w:GD60004] = 32008;          // Display the overlap window to check whether a motor is in operation.
  }

  // Distinguish Operation start and Operation-end after finishing the check mentioned above
  if([b:GB62081] == 1 && [b:GB62082] == 1){
    if([b:GB62085] == 0){          // Prepare Operation start in case the touch panel is not in operation.
      [b:GB62080] = 0;              // End the starting trigger of this script
      [b:GB62083] = 1;              // Operation preparation start trigger
      [b:GB62081] = 0;              // Clear checking flag to see if an alarm is occurring
      [b:GB62082] = 0;              // Clear the In operation checking flag
    } else {                        // Prepare Operation-end in case the touch panel is in operation.
      [b:GB62080] = 0;              // End the starting trigger of this script
      [b:GB62086] = 1;              // Start trigger to terminate operation
      [b:GB62081] = 0;              // Clear checking flag to see if an alarm is occurring
      [b:GB62082] = 0;              // Clear the In operation checking flag
    }
  } else {
    [b:GB62080] = 0;              // End the starting trigger of this script
    [b:GB62081] = 0;              // Clear checking flag to see if an alarm is occurring
    [b:GB62082] = 0;              // Clear the In operation checking flag
  }
}

// 8 axis
if([b:GB62090] == 1){
  [w:GD61000] = 8;

  // Alarm check
  if([1-8:s32:400129] == 0){      // In case of no alarm
    [b:GB62091] = 1;              // Checking flag to see if an alarm is occurring(1: no alarm)
    [w:GD60004] = 0;              // Clear the overlap window for occurring alarm checking.
  } else {                        // in case the alarm occurring
    [b:GB62091] = 0;              // Checking flag to see if an alarm is occurring(0: alarm occurring)
    [w:GD60004] = 32007;          // Display the overlap window for occurring alarm checking.
  }

  // In operation check
  if([1-8:s32:400201] == 0){      // Case: not in operation

```

```

    [b:GB62092] = 1;          // In operation checking flag (1: not in operation)
  } else {                  // In case the motor is running
    [b:GB62092] = 0;          // In operation checking flag (0: in operation)
    [w:GD60004] = 32008;      // Display the overlap window to check whether a motor is in operation.
  }

```

```

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62091] == 1 && [b:GB62092] == 1){
  if([b:GB62095] == 0){    // Prepare Operation start in case the touch panel is not in operation.
    [b:GB62090] = 0;        // End the starting trigger of this script
    [b:GB62093] = 1;        // Operation preparation start trigger
    [b:GB62091] = 0;        // Clear checking flag to see if an alarm is occurring
    [b:GB62092] = 0;        // Clear the In operation checking flag
  } else {                // Prepare Operation-end in case the touch panel is in operation.
    [b:GB62090] = 0;        // End the starting trigger of this script
    [b:GB62096] = 1;        // Start trigger to terminate operation
    [b:GB62091] = 0;        // Clear checking flag to see if an alarm is occurring
    [b:GB62092] = 0;        // Clear the In operation checking flag
  }
} else {
  [b:GB62090] = 0;        // End the starting trigger of this script
  [b:GB62091] = 0;        // Clear checking flag to see if an alarm is occurring
  [b:GB62092] = 0;        // Clear the In operation checking flag
}
}

```

Script No.	31021	Script Name	Script31021
Comment	5-8_Multi operation preparation		
Data Type	Signed BIN16	Trigger Type	ON GB62063

```

// Script to prepare for touch panel operation
// Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter
// Screen script for 5-8 axis multiple operation

// 5-8_Multiple operation preparation script No.31021
// Start the script in case GB62063, GB62073, GB62083, GB62093 in the 5-8_Check before start/end multiple
operation script (No.31020) are ON.

// [b:GB62063], [b:GB62073], [b:GB62083], [b:GB62093] : Operation preparation start trigger (start trigger of
this script)
// [b:GB62064], [b:GB62074], [b:GB62084], [b:GB62094] : Preparing operation flag
// [b:GB62065], [b:GB62075], [b:GB62085], [b:GB62095] : Touch panel in operation flag
// [u32:GD61602], [u32:GD61702], [u32:GD61802], [u32:GD61902] : Backup device for NET-IN0-15
// [u32:GD61634], [u32:GD61734], [u32:GD61834], [u32:GD61934] : Backup device for IN0-7
// [s32:GD61650], [s32:GD61750], [s32:GD61850], [s32:GD61950] : Backup device for Command position of a
monitor
// [s32:GD61652], [s32:GD61752], [s32:GD61852], [u32:GD61952] : Backup device for preset position
// [u32:GD61654],[u32:GD61754],[u32:GD61854],[u32:GD61954] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD61656], [w:GD61756], [w:GD61856], [w:GD61956] : Configuration execution waiting timer

// 5 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62063] == 1 && [b:GB62064] == 0){

  [b:GB62064] = 1;          // Preparing operation flag (1: Preparing operation)
  bmov([1-5:u32:404449],[u32:GD61602],16); // Backs up NET-IN0-15
  bmov([1-5:u32:404353],[u32:GD61634],8);  // Backs up IN0-7
  [s32:GD61652] = [1-5:s32:400909];        // Backs up the preset position
  [u32:GD61654] = [1-5:u32:404169];        // Backup JOG moving distance (used for Minimum JOG moving
distance)

  // Clear the input value of NET-IN
  [1-5:w:400126] = 0;

  // Change the signal assignment for touch panel operation

```

```

[1-5:u32:404449] = 48; // NET-IN0 M0
[1-5:u32:404451] = 49; // NET-IN1 M1
[1-5:u32:404453] = 50; // NET-IN2 M2
[1-5:u32:404455] = 51; // NET-IN3 M3
[1-5:u32:404457] = 52; // NET-IN4 M4
[1-5:u32:404459] = 53; // NET-IN5 M5
[1-5:u32:404461] = 4; // NET-IN6 START
[1-5:u32:404463] = 3; // NET-IN7 HOME
[1-5:u32:404465] = 18; // NET-IN8 STOP
[1-5:u32:404467] = 9; // NET-IN9 MS1
[1-5:u32:404469] = 10; // NET-IN10 MS2
[1-5:u32:404471] = 5; // NET-IN11 SSTART
[1-5:u32:404473] = 6; // NET-IN12 +JOG
[1-5:u32:404475] = 7; // NET-IN13 -JOG
[1-5:u32:404477] = 1; // NET-IN14 FWD
[1-5:u32:404479] = 2; // NET-IN15 RVS

[1-5:u32:404353] = 32; // IN0 R0
[1-5:u32:404355] = 33; // IN1 R1
[1-5:u32:404357] = 34; // IN2 R2
[1-5:u32:404359] = 35; // IN3 R3
[1-5:u32:404361] = 36; // IN4 R4
[1-5:u32:404363] = 37; // IN5 R5
[1-5:u32:404365] = 18; // IN6 STOP
[1-5:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD61650] = [1-5:s32:400199];
[1-5:s32:400909] = [s32:GD61650];

// Execution of Configuration
[1-5:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61656] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61656] > 1 && [b:GB62064] == 1){

    [1-5:s32:400395] = 1; // Executes P-PRESET
    [1-5:s32:400909] = [s32:GD61652]; // Restore the preset position parameter after backup
    [1-5:s32:400397] = 0; // Zero clear execution of Configuration
    [1-5:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62063] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62064] = 0; // Clear Preparing operation flag
    [b:GB62065] = 1; // Touch panel in operation flag (1: in operation)
}

// 6 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62073] == 1 && [b:GB62074] == 0){

    [b:GB62074] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-6:u32:404449],[u32:GD61702],16); // Backs up NET-IN0-15
    bmov([1-6:u32:404353],[u32:GD61734],8); // Backs up IN0-7
    [s32:GD61752] = [1-6:s32:400909]; // Backs up the preset position
    [u32:GD61754] = [1-6:u32:404169]; // Backup JOG moving distance (used for Minimum JOG moving distance)

    // Clear the input value of NET-IN
    [1-6:w:400126] = 0;

```

```

// Change the signal assignment for touch panel operation
[1-6:u32:404449] = 48; // NET-IN0 M0
[1-6:u32:404451] = 49; // NET-IN1 M1
[1-6:u32:404453] = 50; // NET-IN2 M2
[1-6:u32:404455] = 51; // NET-IN3 M3
[1-6:u32:404457] = 52; // NET-IN4 M4
[1-6:u32:404459] = 53; // NET-IN5 M5
[1-6:u32:404461] = 4; // NET-IN6 START
[1-6:u32:404463] = 3; // NET-IN7 HOME
[1-6:u32:404465] = 18; // NET-IN8 STOP
[1-6:u32:404467] = 9; // NET-IN9 MS1
[1-6:u32:404469] = 10; // NET-IN10 MS2
[1-6:u32:404471] = 5; // NET-IN11 SSTART
[1-6:u32:404473] = 6; // NET-IN12 +JOG
[1-6:u32:404475] = 7; // NET-IN13 -JOG
[1-6:u32:404477] = 1; // NET-IN14 FWD
[1-6:u32:404479] = 2; // NET-IN15 RVS

[1-6:u32:404353] = 32; // IN0 R0
[1-6:u32:404355] = 33; // IN1 R1
[1-6:u32:404357] = 34; // IN2 R2
[1-6:u32:404359] = 35; // IN3 R3
[1-6:u32:404361] = 36; // IN4 R4
[1-6:u32:404363] = 37; // IN5 R5
[1-6:u32:404365] = 18; // IN6 STOP
[1-6:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD61750] = [1-6:s32:400199];
[1-6:s32:400909] = [s32:GD61750];

// Execution of Configuration
[1-6:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61756] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61756] > 1 && [b:GB62074] == 1){

    [1-6:s32:400395] = 1; // Executes P-PRESET
    [1-6:s32:400909] = [s32:GD61752]; // Restore the preset position parameter after backup
    [1-6:s32:400397] = 0; // Zero clear execution of Configuration
    [1-6:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62073] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62074] = 0; // Clear Preparing operation flag
    [b:GB62075] = 1; // Touch panel in operation flag (1: in operation)
}

// 7 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62083] == 1 && [b:GB62084] == 0){

    [b:GB62084] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-7:u32:404449],[u32:GD61802],16); // Backs up NET-IN0-15
    bmov([1-7:u32:404353],[u32:GD61834],8); // Backs up IN0-7
    [s32:GD61852] = [1-7:s32:400909]; // Backs up the preset position
    [u32:GD61854] = [1-7:u32:404169]; // Backup JOG moving distance (used for Minimum JOG moving distance)
}

```

```

// Clear the input value of NET-IN
[1-7:w:400126] = 0;

// Change the signal assignment for touch panel operation
[1-7:u32:404449] = 48; // NET-IN0 M0
[1-7:u32:404451] = 49; // NET-IN1 M1
[1-7:u32:404453] = 50; // NET-IN2 M2
[1-7:u32:404455] = 51; // NET-IN3 M3
[1-7:u32:404457] = 52; // NET-IN4 M4
[1-7:u32:404459] = 53; // NET-IN5 M5
[1-7:u32:404461] = 4; // NET-IN6 START
[1-7:u32:404463] = 3; // NET-IN7 HOME
[1-7:u32:404465] = 18; // NET-IN8 STOP
[1-7:u32:404467] = 9; // NET-IN9 MS1
[1-7:u32:404469] = 10; // NET-IN10 MS2
[1-7:u32:404471] = 5; // NET-IN11 SSTART
[1-7:u32:404473] = 6; // NET-IN12 +JOG
[1-7:u32:404475] = 7; // NET-IN13 -JOG
[1-7:u32:404477] = 1; // NET-IN14 FWD
[1-7:u32:404479] = 2; // NET-IN15 RVS

[1-7:u32:404353] = 32; // IN0 R0
[1-7:u32:404355] = 33; // IN1 R1
[1-7:u32:404357] = 34; // IN2 R2
[1-7:u32:404359] = 35; // IN3 R3
[1-7:u32:404361] = 36; // IN4 R4
[1-7:u32:404363] = 37; // IN5 R5
[1-7:u32:404365] = 18; // IN6 STOP
[1-7:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD61850] = [1-7:s32:400199];
[1-7:s32:400909] = [s32:GD61850];

// Execution of Configuration
[1-7:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD61856] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61856] > 1 && [b:GB62084] == 1){

    [1-7:s32:400395] = 1; // Executes P-PRESET
    [1-7:s32:400909] = [s32:GD61852]; // Restore the preset position parameter after backup
    [1-7:s32:400397] = 0; // Zero clear execution of Configuration
    [1-7:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62083] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62084] = 0; // Clear Preparing operation flag
    [b:GB62085] = 1; // Touch panel in operation flag (1: in operation)
}

// 8 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62093] == 1 && [b:GB62094] == 0){

    [b:GB62094] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-8:u32:404449],[u32:GD61902],16); // Backs up NET-IN0-15
    bmov([1-8:u32:404353],[u32:GD61934],8); // Backs up IN0-7
    [s32:GD61952] = [1-8:s32:400909]; // Backs up the preset position
    [u32:GD61954] = [1-8:u32:404169]; // Backup JOG moving distance (used for Minimum JOG moving

```

distance)

```
// Clear the input value of NET-IN
[1-8:w:400126] = 0;
```

```
// Change the signal assignment for touch panel operation
```

```
[1-8:u32:404449] = 48; // NET-IN0 M0
[1-8:u32:404451] = 49; // NET-IN1 M1
[1-8:u32:404453] = 50; // NET-IN2 M2
[1-8:u32:404455] = 51; // NET-IN3 M3
[1-8:u32:404457] = 52; // NET-IN4 M4
[1-8:u32:404459] = 53; // NET-IN5 M5
[1-8:u32:404461] = 4; // NET-IN6 START
[1-8:u32:404463] = 3; // NET-IN7 HOME
[1-8:u32:404465] = 18; // NET-IN8 STOP
[1-8:u32:404467] = 9; // NET-IN9 MS1
[1-8:u32:404469] = 10; // NET-IN10 MS2
[1-8:u32:404471] = 5; // NET-IN11 SSTART
[1-8:u32:404473] = 6; // NET-IN12 +JOG
[1-8:u32:404475] = 7; // NET-IN13 -JOG
[1-8:u32:404477] = 1; // NET-IN14 FWD
[1-8:u32:404479] = 2; // NET-IN15 RVS
```

```
[1-8:u32:404353] = 32; // IN0 R0
[1-8:u32:404355] = 33; // IN1 R1
[1-8:u32:404357] = 34; // IN2 R2
[1-8:u32:404359] = 35; // IN3 R3
[1-8:u32:404361] = 36; // IN4 R4
[1-8:u32:404363] = 37; // IN5 R5
[1-8:u32:404365] = 18; // IN6 STOP
[1-8:u32:404367] = 39; // IN7 R7
```

```
// Set the preset position parameter to the command position of the monitor
[s32:GD61950] = [1-8:s32:400199];
[1-8:s32:400909] = [s32:GD61950];
```

```
// Execution of Configuration
[1-8:s32:400397] = 1;
```

```
// Start recording of Configuration execution waiting timer [sec]
[w:GD61956] = [w:GS7];
```

```
}
```

```
// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD61956] > 1 && [b:GB62094] == 1){
```

```
[1-8:s32:400395] = 1; // Executes P-PRESET
[1-8:s32:400909] = [s32:GD61952]; // Restore the preset position parameter after backup
[1-8:s32:400397] = 0; // Zero clear execution of Configuration
[1-8:s32:400395] = 0; // Zero clear for the execution of P-PRESET
```

```
[b:GB62093] = 0; // Clear Operation preparation start trigger
// End the starting trigger in this script.
[b:GB62094] = 0; // Clear Preparing operation flag
[b:GB62095] = 1; // Touch panel in operation flag (1: in operation)
```

```
}
```

Script No.	31022	Script Name	Script31022
Comment	5-8_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62065

```
// Script for positioning operation with touch panel operation
// Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No.
// After being changed, M0-M5 are turned ON by driver input command (400126).
```

```
// 5-8_Script in multiple operation No.31022
```



```

// Start the script in case GB62065, GB62075, GB62085, GB62095 in the 5-8_multiple operation preparation
script (No.31021) are ON.

// [b:GB62065],[b:GB62075],[b:GB62085],[b:GB62095] : Touch panel in operation flag(start trigger of this script)
// Bit also works as interlock not to change the screen while touch panel in operation

// [1-5:u32:400195] - [1-8:u32:400195] : Modbus address of selected data No.
// [u32:GD61600], [u32:GD61700], [GD61800], [u32:GD61900] : Device for the operation data No.

// Change M0-M5 of driver input command to reflect the selected operation data No.
// To ease the touch panel processing load,
// In case the currently selected data No. (400195) of the monitor and the selected operation data No. are not
the same,
// change the driver input command (400126) to reflect a change

// 5 axis
if([b:GB62065] == 1){

    if([1-5:u32:400195] != [u32:GD61600]){

        [1-5:b:400126.b0] = [b:GD61600.b0];    // M0
        [1-5:b:400126.b1] = [b:GD61600.b1];    // M1
        [1-5:b:400126.b2] = [b:GD61600.b2];    // M2
        [1-5:b:400126.b3] = [b:GD61600.b3];    // M3
        [1-5:b:400126.b4] = [b:GD61600.b4];    // M4
        [1-5:b:400126.b5] = [b:GD61600.b5];    // M5
    }
}

// 6 axis
if([b:GB62075] == 1){

    if([1-6:u32:400195] != [u32:GD61700]){

        [1-6:b:400126.b0] = [b:GD61700.b0];    // M0
        [1-6:b:400126.b1] = [b:GD61700.b1];    // M1
        [1-6:b:400126.b2] = [b:GD61700.b2];    // M2
        [1-6:b:400126.b3] = [b:GD61700.b3];    // M3
        [1-6:b:400126.b4] = [b:GD61700.b4];    // M4
        [1-6:b:400126.b5] = [b:GD61700.b5];    // M5
    }
}

// 7 axis
if([b:GB62085] == 1){

    if([1-7:u32:400195] != [u32:GD61800]){

        [1-7:b:400126.b0] = [b:GD61800.b0];    // M0
        [1-7:b:400126.b1] = [b:GD61800.b1];    // M1
        [1-7:b:400126.b2] = [b:GD61800.b2];    // M2
        [1-7:b:400126.b3] = [b:GD61800.b3];    // M3
        [1-7:b:400126.b4] = [b:GD61800.b4];    // M4
        [1-7:b:400126.b5] = [b:GD61800.b5];    // M5
    }
}

// 8 axis
if([b:GB62095] == 1){

    if([1-8:u32:400195] != [u32:GD61900]){

        [1-8:b:400126.b0] = [b:GD61900.b0];    // M0
        [1-8:b:400126.b1] = [b:GD61900.b1];    // M1
        [1-8:b:400126.b2] = [b:GD61900.b2];    // M2
        [1-8:b:400126.b3] = [b:GD61900.b3];    // M3
        [1-8:b:400126.b4] = [b:GD61900.b4];    // M4
        [1-8:b:400126.b5] = [b:GD61900.b5];    // M5
    }
}

```

<pre> } } </pre>			
Script No.	31023	Script Name	Script31023
Comment	5-8_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62066
<pre> // Script to disable touch panel operation // Restore such as the NET-IN signal and the necessary parameter from backup // Screen script for 5-8 axis multiple operation  // 5-8_End multiple operation script No.31023 // Start the script in case GB62066, GB62076, GB62086, GB62096 in the 5-8_Check before start/end multiple operation script (No.31020) are ON.  // [b:GB62065], [b:GB62075], [b:GB62085], [b:GB62095] : Touch panel in operation flag // [b:GB62066], [b:GB62076], [b:GB62086], [b:GB62096] : Operation-end start trigger (start trigger of this script) // [b:GB62067], [b:GB62077], [b:GB62087], [b:GB62097] : the operation-end in process flag // [u32:GD61602], [u32:GD61702], [u32:GD61802], [u32:GD61902] : Backup device for NET-IN0-15 // [u32:GD61634], [u32:GD61734], [u32:GD61834], [u32:GD61934] : Backup device for IN0-7 // [s32:GD61650], [s32:GD61750], [s32:GD61850], [s32:GD61950] : Backup device for Command position of a monitor // [s32:GD61652], [s32:GD61752], [s32:GD61852], [s32:GD61952] : Backup device for preset position // [u32:GD61654],[u32:GD61754],[u32:GD61854],[u32:GD61954] : Backup device for JOG moving distance (minimum moving distance for GOT) // [w:GD61656], [w:GD61756], [w:GD61856], [w:GD61956] : Configuration execution waiting timer  // 5 axis // In case Operation-end start trigger and Operation-end are not completed. if([b:GB62066] == 1 &amp;&amp; [b:GB62067] == 0){      [b:GB62065] = 0;                // Clear touch panel in operation flag     [b:GB62067] = 1;                // the operation-end in process flag (1: operation end in process)     bmov([u32:GD61602],[1-5:u32:404449],16); // Restore NET-IN0-15 from backup     bmov([u32:GD61634],[1-5:u32:404353],8);  // Restore IN0-7 from backup     [1-5:u32:404169] = [u32:GD61654];        // Restore JOG moving distance from backup      // Set the preset position parameter to the command position of the monitor     [s32:GD61650] = [1-5:s32:400199];     [1-5:s32:400909] = [s32:GD61650];      // Clear the input value of NET-IN     [1-5:w:400126] = 0;      // Execution of Configuration     [1-5:s32:400397] = 1;      // Start recording of Configuration execution waiting timer [sec]     [w:GD61656] = [w:GS7]; }  // In case Configuration execution and Operation-end are completed. if([w:GS7] - [w:GD61656] &gt; 1 &amp;&amp; [b:GB62067] == 1){      [1-5:s32:400395] = 1;            // Executes P-PRESET     [1-5:s32:400909] = [s32:GD61652]; // Reset the Preset position parameter     [1-5:s32:400397] = 0;            // Zero clear execution of Configuration     [1-5:s32:400395] = 0;            // Zero clear for the execution of P-PRESET      [b:GB62066] = 0;                // Clear Operation-end start trigger     // End the starting trigger in this script.     [b:GB62067] = 0;                // Clear the operation-end in process flag }  // 6 axis </pre>			

```

// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62076] == 1 && [b:GB62077] == 0){

    [b:GB62075] = 0;                // Clear touch panel in operation flag
    [b:GB62077] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD61702],[1-6:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD61734],[1-6:u32:404353],8);  // Restore IN0-7 from backup
    [1-6:u32:404169] = [u32:GD61754];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD61750] = [1-6:s32:400199];
    [1-6:s32:400909] = [s32:GD61750];

    // Clear the input value of NET-IN
    [1-6:w:400126] = 0;

    // Execution of Configuration
    [1-6:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD61756] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD61756] > 1 && [b:GB62077] == 1){

    [1-6:s32:400395] = 1;          // Executes P-PRESET
    [1-6:s32:400909] = [s32:GD61752]; // Reset the Preset position parameter
    [1-6:s32:400397] = 0;          // Zero clear execution of Configuration
    [1-6:s32:400395] = 0;          // Zero clear for the execution of P-PRESET

    [b:GB62076] = 0;              // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62077] = 0;              // Clear the operation-end in process flag
}

// 7 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62086] == 1 && [b:GB62087] == 0){

    [b:GB62085] = 0;                // Clear touch panel in operation flag
    [b:GB62087] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD61802],[1-7:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD61834],[1-7:u32:404353],8);  // Restore IN0-7 from backup
    [1-7:u32:404169] = [u32:GD61854];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD61850] = [1-7:s32:400199];
    [1-7:s32:400909] = [s32:GD61850];

    // Clear the input value of NET-IN
    [1-7:w:400126] = 0;

    // Execution of Configuration
    [1-7:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD61856] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD61856] > 1 && [b:GB62087] == 1){

    [1-7:s32:400395] = 1;          // Executes P-PRESET

```

```

[1-7:s32:400909] = [s32:GD61852]; // Reset the Preset position parameter
[1-7:s32:400397] = 0;           // Zero clear execution of Configuration
[1-7:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

[b:GB62086] = 0;                // Clear Operation-end start trigger
                                // End the starting trigger in this script.
[b:GB62087] = 0;                // Clear the operation-end in process flag
}

// 8 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62096] == 1 && [b:GB62097] == 0){

    [b:GB62095] = 0;             // Clear touch panel in operation flag
    [b:GB62097] = 1;             // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD61902],[1-8:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD61934],[1-8:u32:404353],8);  // Restore IN0-7 from backup
    [1-8:u32:404169] = [u32:GD61954];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD61950] = [1-8:s32:400199];
    [1-8:s32:400909] = [s32:GD61950];

    // Clear the input value of NET-IN
    [1-8:w:400126] = 0;

    // Execution of Configuration
    [1-8:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD61956] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD61956] > 1 && [b:GB62097] == 1){

    [1-8:s32:400395] = 1;        // Executes P-PRESET
    [1-8:s32:400909] = [s32:GD61952]; // Reset the Preset position parameter
    [1-8:s32:400397] = 0;        // Zero clear execution of Configuration
    [1-8:s32:400395] = 0;        // Zero clear for the execution of P-PRESET

    [b:GB62096] = 0;            // Clear Operation-end start trigger
                                // End the starting trigger in this script.
    [b:GB62097] = 0;            // Clear the operation-end in process flag
}

```

Script No.	31020	Script Name	Script31020
Comment	5-8ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62070
Same as the script No. 31020 for the base screen 31023			
Script No.	31021	Script Name	Script31021
Comment	5-8_Multi operation preparation		
Data Type	Signed BIN16	Trigger Type	ON GB62073
Same as the script No. 31021 for the base screen 31023			
Script No.	31022	Script Name	Script31022
Comment	5-8_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62075
Same as the script No. 31022 for the base screen 31023			
Script No.	31023	Script Name	Script31023

Comment	5-8_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62076
Same as the script No. 31023 for the base screen 31023			
Script No.	31020	Script Name	Script31020
Comment	5-8ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62080
Same as the script No. 31020 for the base screen 31023			
Script No.	31021	Script Name	Script31021
Comment	5-8_Multi operation preparation		
Data Type	Signed BIN16	Trigger Type	ON GB62083
Same as the script No. 31021 for the base screen 31023			
Script No.	31022	Script Name	Script31022
Comment	5-8_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62085
Same as the script No. 31022 for the base screen 31023			
Script No.	31023	Script Name	Script31023
Comment	5-8_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62086
Same as the script No. 31023 for the base screen 31023			
Script No.	31020	Script Name	Script31020
Comment	5-8ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62090
Same as the script No. 31020 for the base screen 31023			
Script No.	31021	Script Name	Script31021
Comment	5-8_Multi operation preparation		
Data Type	Signed BIN16	Trigger Type	ON GB62093
Same as the script No. 31021 for the base screen 31023			
Script No.	31022	Script Name	Script31022
Comment	5-8_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62095
Same as the script No. 31022 for the base screen 31023			
Script No.	31023	Script Name	Script31023
Comment	5-8_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62096
Same as the script No. 31023 for the base screen 31023			

#### Base screen 31024

Script No.	31024	Script Name	Script31024
Comment	9-12ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62100
// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end // Screen script for 9-12 axis multiple operation  // 9-12_Check before start/end multiple operation script No.31024 // Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of			

## 9-12 axis

```
// [b:GB62100] : 9 axis starting trigger using bit set
// [b:GB62110] : 10 axis starting trigger using bit set
// [b:GB62120] : 11 axis starting trigger using bit set
// [b:GB62130] : 12 axis starting trigger using bit set
// [b:GB62105] : 9 axis touch panel in operation flag
// [b:GB62115] : 10 axis touch panel in operation flag
// [b:GB62125] : 11 axis touch panel in operation flag
// [b:GB62135] : 12 axis touch panel in operation flag

// [w:GD61000] = 1 means as below
// Because Occurring alarm checking screen is responsible for station No. switching
// Station No.(CPU No. of driver) is needed for alarm clear (Therefore settings of each axis is required.)
// Screen switching device (GD61199) is responsible for Station No. reset

// 9 axis
if([b:GB62100] == 1){
    [w:GD61000] = 9;

    // Alarm check
    if([1-9:s32:400129] == 0){ // In case of no alarm
        [b:GB62101] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62101] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-9:s32:400201] == 0){ // Case: not in operation
        [b:GB62102] = 1; // In operation checking flag (1: not in operation)
    } else { // In case the motor is running
        [b:GB62102] = 0; // In operation checking flag (0: in operation)
        [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62101] == 1 && [b:GB62102] == 1){
        if([b:GB62105] == 0){ // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62100] = 0; // End the starting trigger of this script
            [b:GB62103] = 1; // Operation preparation start trigger
            [b:GB62101] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62102] = 0; // Clear the In operation checking flag
        } else { // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62100] = 0; // End the starting trigger of this script
            [b:GB62106] = 1; // Start trigger to terminate operation
            [b:GB62101] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62102] = 0; // Clear the In operation checking flag
        }
    } else {
        [b:GB62100] = 0; // End the starting trigger of this script
        [b:GB62101] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62102] = 0; // Clear the In operation checking flag
    }
}

// 10 axis
if([b:GB62110] == 1){
    [w:GD61000] = 10;

    // Alarm check
    if([1-10:s32:400129] == 0){ // In case of no alarm
        [b:GB62111] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
```

```

[w:GD60004] = 0;      // Clear the overlap window for occurring alarm checking.
} else {              // in case the alarm occurring
    [b:GB62111] = 0;    // Checking flag to see if an alarm is occurring(0: alarm occurring)
    [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
}

// In operation check
if([1-10:s32:400201] == 0){ // Case: not in operation
    [b:GB62112] = 1;        // In operation checking flag (1: not in operation)
} else {                    // In case the motor is running
    [b:GB62112] = 0;        // In operation checking flag (0: in operation)
    [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62111] == 1 && [b:GB62112] == 1){
    if([b:GB62115] == 0){ // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62110] = 0; // End the starting trigger of this script
        [b:GB62113] = 1; // Operation preparation start trigger
        [b:GB62111] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62112] = 0; // Clear the In operation checking flag
    } else { // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62110] = 0; // End the starting trigger of this script
        [b:GB62116] = 1; // Start trigger to terminate operation
        [b:GB62111] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62112] = 0; // Clear the In operation checking flag
    }
} else {
    [b:GB62110] = 0; // End the starting trigger of this script
    [b:GB62111] = 0; // Clear checking flag to see if an alarm is occurring
    [b:GB62112] = 0; // Clear the In operation checking flag
}
}

// 11 axis
if([b:GB62120] == 1){
    [w:GD61000] = 11;

    // Alarm check
    if([1-11:s32:400129] == 0){ // In case of no alarm
        [b:GB62121] = 1;        // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;        // Clear the overlap window for occurring alarm checking.
    } else {                    // in case the alarm occurring
        [b:GB62121] = 0;        // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-11:s32:400201] == 0){ // Case: not in operation
        [b:GB62122] = 1;        // In operation checking flag (1: not in operation)
    } else {                    // In case the motor is running
        [b:GB62122] = 0;        // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62121] == 1 && [b:GB62122] == 1){
        if([b:GB62125] == 0){ // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62120] = 0; // End the starting trigger of this script
            [b:GB62123] = 1; // Operation preparation start trigger
            [b:GB62121] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62122] = 0; // Clear the In operation checking flag
        } else { // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62120] = 0; // End the starting trigger of this script

```

```

        [b:GB62126] = 1;    // Start trigger to terminate operation
        [b:GB62121] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62122] = 0;    // Clear the In operation checking flag
    }
} else {
    [b:GB62120] = 0;    // End the starting trigger of this script
    [b:GB62121] = 0;    // Clear checking flag to see if an alarm is occurring
    [b:GB62122] = 0;    // Clear the In operation checking flag
}
}

// 12 axis
if([b:GB62130] == 1){
    [w:GD61000] = 12;

    // Alarm check
    if([1-12:s32:400129] == 0){    // In case of no alarm
        [b:GB62131] = 1;    // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;    // Clear the overlap window for occurring alarm checking.
    } else {    // in case the alarm occurring
        [b:GB62131] = 0;    // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-12:s32:400201] == 0){    // Case: not in operation
        [b:GB62132] = 1;    // In operation checking flag (1: not in operation)
    } else {    // In case the motor is running
        [b:GB62132] = 0;    // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62131] == 1 && [b:GB62132] == 1){
        if([b:GB62135] == 0){    // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62130] = 0;    // End the starting trigger of this script
            [b:GB62133] = 1;    // Operation preparation start trigger
            [b:GB62131] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62132] = 0;    // Clear the In operation checking flag
        } else {    // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62130] = 0;    // End the starting trigger of this script
            [b:GB62136] = 1;    // Start trigger to terminate operation
            [b:GB62131] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62132] = 0;    // Clear the In operation checking flag
        }
    } else {
        [b:GB62130] = 0;    // End the starting trigger of this script
        [b:GB62131] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62132] = 0;    // Clear the In operation checking flag
    }
}
}

```

Script No.	31025	Script Name	Script31025
Comment	9-12_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62103

// Script to prepare for touch panel operation  
 // Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter  
 // Screen script for 9-12 axis multiple operation

// 9-12\_Multiple operation preparation script No.31025  
 // Start the script in case GB62103, GB62113, GB62123, GB62133 in the 9-12\_Check before start/end multiple operation script (No.31024) are ON.

// [b:GB62103], [b:GB62113], [b:GB62123], [b:GB62133] : Operation preparation start trigger (start trigger of this script)



```

// [b:GB62104], [b:GB62114], [b:GB62124], [b:GB62134] : Preparing operation flag
// [b:GB62105], [b:GB62115], [b:GB62125], [b:GB62135] : Touch panel in operation flag
// [u32:GD62002], [u32:GD62102], [u32:GD62202], [u32:GD62302] : Backup device for NET-IN0-15
// [u32:GD62034], [u32:GD62134], [u32:GD62234], [u32:GD62334] : Backup device for IN0-7
// [s32:GD61650], [s32:GD61750], [s32:GD61850], [s32:GD61950] : Backup device for Command position of
a monitor
// [s32:GD62052], [s32:GD62152], [s32:GD62252], [u32:GD62352] : Backup device for preset position
// [u32:GD62054],[u32:GD62154],[u32:GD62254],[u32:GD62354] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD62056], [w:GD62156], [w:GD62256], [w:GD62356] : Configuration execution waiting timer

// 9 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62103] == 1 && [b:GB62104] == 0){

    [b:GB62104] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-9:u32:404449],[u32:GD62002],16); // Backs up NET-IN0-15
    bmov([1-9:u32:404353],[u32:GD62034],8); // Backs up IN0-7
    [s32:GD62052] = [1-9:s32:400909]; // Backs up the preset position
    [u32:GD62054] = [1-9:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-9:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-9:u32:404449] = 48; // NET-IN0 M0
    [1-9:u32:404451] = 49; // NET-IN1 M1
    [1-9:u32:404453] = 50; // NET-IN2 M2
    [1-9:u32:404455] = 51; // NET-IN3 M3
    [1-9:u32:404457] = 52; // NET-IN4 M4
    [1-9:u32:404459] = 53; // NET-IN5 M5
    [1-9:u32:404461] = 4; // NET-IN6 START
    [1-9:u32:404463] = 3; // NET-IN7 HOME
    [1-9:u32:404465] = 18; // NET-IN8 STOP
    [1-9:u32:404467] = 9; // NET-IN9 MS1
    [1-9:u32:404469] = 10; // NET-IN10 MS2
    [1-9:u32:404471] = 5; // NET-IN11 SSTART
    [1-9:u32:404473] = 6; // NET-IN12 +JOG
    [1-9:u32:404475] = 7; // NET-IN13 -JOG
    [1-9:u32:404477] = 1; // NET-IN14 FWD
    [1-9:u32:404479] = 2; // NET-IN15 RVS

    [1-9:u32:404353] = 32; // IN0 R0
    [1-9:u32:404355] = 33; // IN1 R1
    [1-9:u32:404357] = 34; // IN2 R2
    [1-9:u32:404359] = 35; // IN3 R3
    [1-9:u32:404361] = 36; // IN4 R4
    [1-9:u32:404363] = 37; // IN5 R5
    [1-9:u32:404365] = 18; // IN6 STOP
    [1-9:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62050] = [1-9:s32:400199];
    [1-9:s32:400909] = [s32:GD62050];

    // Execution of Configuration
    [1-9:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62056] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.

```

```

if([w:GS7] - [w:GD62056] > 1 && [b:GB62104] == 1){

    [1-9:s32:400395] = 1;           // Executes P-PRESET
    [1-9:s32:400909] = [s32:GD62052]; // Restore the preset position parameter after backup
    [1-9:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-9:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62103] = 0;               // Clear Operation preparation start trigger
                                   // End the starting trigger in this script.
    [b:GB62104] = 0;               // Clear Preparing operation flag
    [b:GB62105] = 1;               // Touch panel in operation flag (1: in operation)
}

// 10 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62113] == 1 && [b:GB62114] == 0){

    [b:GB62114] = 1;               // Preparing operation flag (1: Preparing operation)
    bmov([1-10:u32:404449],[u32:GD62102],16); // Backs up NET-IN0-15
    bmov([1-10:u32:404353],[u32:GD62134],8);  // Backs up IN0-7
    [s32:GD62152] = [1-10:s32:400909];        // Backs up the preset position
    [u32:GD62154] = [1-10:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-10:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-10:u32:404449] = 48; // NET-IN0  M0
    [1-10:u32:404451] = 49; // NET-IN1  M1
    [1-10:u32:404453] = 50; // NET-IN2  M2
    [1-10:u32:404455] = 51; // NET-IN3  M3
    [1-10:u32:404457] = 52; // NET-IN4  M4
    [1-10:u32:404459] = 53; // NET-IN5  M5
    [1-10:u32:404461] = 4;  // NET-IN6  START
    [1-10:u32:404463] = 3;  // NET-IN7  HOME
    [1-10:u32:404465] = 18; // NET-IN8  STOP
    [1-10:u32:404467] = 9;  // NET-IN9  MS1
    [1-10:u32:404469] = 10; // NET-IN10 MS2
    [1-10:u32:404471] = 5;  // NET-IN11 SSTART
    [1-10:u32:404473] = 6;  // NET-IN12 +JOG
    [1-10:u32:404475] = 7;  // NET-IN13 -JOG
    [1-10:u32:404477] = 1;  // NET-IN14 FWD
    [1-10:u32:404479] = 2;  // NET-IN15 RVS

    [1-10:u32:404353] = 32; // IN0 R0
    [1-10:u32:404355] = 33; // IN1 R1
    [1-10:u32:404357] = 34; // IN2 R2
    [1-10:u32:404359] = 35; // IN3 R3
    [1-10:u32:404361] = 36; // IN4 R4
    [1-10:u32:404363] = 37; // IN5 R5
    [1-10:u32:404365] = 18; // IN6 STOP
    [1-10:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62150] = [1-10:s32:400199];
    [1-10:s32:400909] = [s32:GD62150];

    // Execution of Configuration
    [1-10:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62156] = [w:GS7];
}

```

```

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62156] > 1 && [b:GB62114] == 1){

    [1-10:s32:400395] = 1;           // Executes P-PRESET
    [1-10:s32:400909] = [s32:GD62152]; // Restore the preset position parameter after backup
    [1-10:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-10:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62113] = 0;                // Clear Operation preparation start trigger
                                    // End the starting trigger in this script.
    [b:GB62114] = 0;                // Clear Preparing operation flag
    [b:GB62115] = 1;                // Touch panel in operation flag (1: in operation)
}

// 11 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62123] == 1 && [b:GB62124] == 0){

    [b:GB62124] = 1;                // Preparing operation flag (1: Preparing operation)
    bmov([1-11:u32:404449],[u32:GD62202],16); // Backs up NET-IN0-15
    bmov([1-11:u32:404353],[u32:GD62234],8);  // Backs up IN0-7
    [s32:GD62252] = [1-11:s32:400909];        // Backs up the preset position
    [u32:GD62254] = [1-11:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-11:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-11:u32:404449] = 48; // NET-IN0 M0
    [1-11:u32:404451] = 49; // NET-IN1 M1
    [1-11:u32:404453] = 50; // NET-IN2 M2
    [1-11:u32:404455] = 51; // NET-IN3 M3
    [1-11:u32:404457] = 52; // NET-IN4 M4
    [1-11:u32:404459] = 53; // NET-IN5 M5
    [1-11:u32:404461] = 4;  // NET-IN6 START
    [1-11:u32:404463] = 3;  // NET-IN7 HOME
    [1-11:u32:404465] = 18; // NET-IN8 STOP
    [1-11:u32:404467] = 9;  // NET-IN9 MS1
    [1-11:u32:404469] = 10; // NET-IN10 MS2
    [1-11:u32:404471] = 5;  // NET-IN11 SSTART
    [1-11:u32:404473] = 6;  // NET-IN12 +JOG
    [1-11:u32:404475] = 7;  // NET-IN13 -JOG
    [1-11:u32:404477] = 1;  // NET-IN14 FWD
    [1-11:u32:404479] = 2;  // NET-IN15 RVS

    [1-11:u32:404353] = 32; // IN0 R0
    [1-11:u32:404355] = 33; // IN1 R1
    [1-11:u32:404357] = 34; // IN2 R2
    [1-11:u32:404359] = 35; // IN3 R3
    [1-11:u32:404361] = 36; // IN4 R4
    [1-11:u32:404363] = 37; // IN5 R5
    [1-11:u32:404365] = 18; // IN6 STOP
    [1-11:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62250] = [1-11:s32:400199];
    [1-11:s32:400909] = [s32:GD62250];

    // Execution of Configuration
    [1-11:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]

```

```

[w:GD62256] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62256] > 1 && [b:GB62124] == 1){

    [1-11:s32:400395] = 1;           // Executes P-PRESET
    [1-11:s32:400909] = [s32:GD62252]; // Restore the preset position parameter after backup
    [1-11:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-11:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62123] = 0;                // Clear Operation preparation start trigger
                                    // End the starting trigger in this script.
    [b:GB62124] = 0;                // Clear Preparing operation flag
    [b:GB62125] = 1;                // Touch panel in operation flag (1: in operation)
}

// 12 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62133] == 1 && [b:GB62134] == 0){

    [b:GB62134] = 1;                // Preparing operation flag (1: Preparing operation)
    bmov([1-12:u32:404449],[u32:GD62302],16); // Backs up NET-IN0-15
    bmov([1-12:u32:404353],[u32:GD62334],8);  // Backs up IN0-7
    [s32:GD62352] = [1-12:s32:400909];        // Backs up the preset position
    [u32:GD62354] = [1-12:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-12:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-12:u32:404449] = 48; // NET-IN0 M0
    [1-12:u32:404451] = 49; // NET-IN1 M1
    [1-12:u32:404453] = 50; // NET-IN2 M2
    [1-12:u32:404455] = 51; // NET-IN3 M3
    [1-12:u32:404457] = 52; // NET-IN4 M4
    [1-12:u32:404459] = 53; // NET-IN5 M5
    [1-12:u32:404461] = 4;  // NET-IN6 START
    [1-12:u32:404463] = 3;  // NET-IN7 HOME
    [1-12:u32:404465] = 18; // NET-IN8 STOP
    [1-12:u32:404467] = 9;  // NET-IN9 MS1
    [1-12:u32:404469] = 10; // NET-IN10 MS2
    [1-12:u32:404471] = 5;  // NET-IN11 SSTART
    [1-12:u32:404473] = 6;  // NET-IN12 +JOG
    [1-12:u32:404475] = 7;  // NET-IN13 -JOG
    [1-12:u32:404477] = 1;  // NET-IN14 FWD
    [1-12:u32:404479] = 2;  // NET-IN15 RVS

    [1-12:u32:404353] = 32; // IN0 R0
    [1-12:u32:404355] = 33; // IN1 R1
    [1-12:u32:404357] = 34; // IN2 R2
    [1-12:u32:404359] = 35; // IN3 R3
    [1-12:u32:404361] = 36; // IN4 R4
    [1-12:u32:404363] = 37; // IN5 R5
    [1-12:u32:404365] = 18; // IN6 STOP
    [1-12:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62350] = [1-12:s32:400199];
    [1-12:s32:400909] = [s32:GD62350];

    // Execution of Configuration
    [1-12:s32:400397] = 1;

```

```

// Start recording of Configuration execution waiting timer [sec]
[w:GD62356] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62356] > 1 && [b:GB62134] == 1){

    [1-12:s32:400395] = 1;           // Executes P-PRESET
    [1-12:s32:400909] = [s32:GD62352]; // Restore the preset position parameter after backup
    [1-12:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-12:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62133] = 0;                // Clear Operation preparation start trigger
                                    // End the starting trigger in this script.
    [b:GB62134] = 0;                // Clear Preparing operation flag
    [b:GB62135] = 1;                // Touch panel in operation flag (1: in operation)
}

```

Script No.	31026	Script Name	Script31026
Comment	9-12_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62105

```

// Script for positioning operation with touch panel operation
// Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No.
// After being changed, M0-M5 are turned ON by driver input command (400126).

// 9-12_Script in multiple operation No.31026
// Start the script in case GB62105, GB62115, GB62125, GB62135 in the 9-12_multiple operation preparation
script (No.31025) are ON.

// [b:GB62105],[b:GB62115],[b:GB62125],[b:GB62135] : Touch panel in operation flag(start trigger of this
script)
// Bit also works as interlock not to change the screen while touch panel in operation

// [1-9:u32:400195] - [1-12:u32:400195] : Modbus address of selected data No.
// [u32:GD62000], [u32GD62100], [GD62200], [u32:GD62300] : Device for the operation data No.

// Change M0-M5 of driver input command to reflect the selected operation data No.
// To ease the touch panel processing load,
// In case the currently selected data No. (400915) of the monitor and the selected operation data No. are not
the same,
// change the driver input command (400126) to reflect a change

// 9 axis
if([b:GB62105] == 1){

    if([1-9:u32:400195] != [u32:GD62000]){
        [1-9:b:400126.b0] = [b:GD62000.b0]; // M0
        [1-9:b:400126.b1] = [b:GD62000.b1]; // M1
        [1-9:b:400126.b2] = [b:GD62000.b2]; // M2
        [1-9:b:400126.b3] = [b:GD62000.b3]; // M3
        [1-9:b:400126.b4] = [b:GD62000.b4]; // M4
        [1-9:b:400126.b5] = [b:GD62000.b5]; // M5
    }
}

// 10 axis
if([b:GB62115] == 1){

    if([1-10:u32:400195] != [u32:GD62100]){
        [1-10:b:400126.b0] = [b:GD62100.b0]; // M0
        [1-10:b:400126.b1] = [b:GD62100.b1]; // M1
        [1-10:b:400126.b2] = [b:GD62100.b2]; // M2
        [1-10:b:400126.b3] = [b:GD62100.b3]; // M3
        [1-10:b:400126.b4] = [b:GD62100.b4]; // M4
    }
}

```

```

    [1-10:b:400126.b5] = [b:GD62100.b5]; // M5
  }
}

// 11 axis
if([b:GB62125] == 1){

  if([1-11:u32:400195] != [u32:GD62200]){
    [1-11:b:400126.b0] = [b:GD62200.b0]; // M0
    [1-11:b:400126.b1] = [b:GD62200.b1]; // M1
    [1-11:b:400126.b2] = [b:GD62200.b2]; // M2
    [1-11:b:400126.b3] = [b:GD62200.b3]; // M3
    [1-11:b:400126.b4] = [b:GD62200.b4]; // M4
    [1-11:b:400126.b5] = [b:GD62200.b5]; // M5
  }
}

// 12 axis
if([b:GB62135] == 1){

  if([1-12:u32:400195] != [u32:GD62300]){
    [1-12:b:400126.b0] = [b:GD62300.b0]; // M0
    [1-12:b:400126.b1] = [b:GD62300.b1]; // M1
    [1-12:b:400126.b2] = [b:GD62300.b2]; // M2
    [1-12:b:400126.b3] = [b:GD62300.b3]; // M3
    [1-12:b:400126.b4] = [b:GD62300.b4]; // M4
    [1-12:b:400126.b5] = [b:GD62300.b5]; // M5
  }
}
}

```

Script No.	31027	Script Name	Script31027
Comment	9-12_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62106

```

// Script to disable touch panel operation
// Restore such as the NET-IN signal and the necessary parameter from backup
// Screen script for 9-12 axis multiple operation

// 9-12_End multiple operation script No.31027
// Start the script in case GB62106, GB62116, GB62126, GB62136 in the 9-12_Check before start/end
multiple operation script (No.31024) are ON.

// [b:GB62105], [b:GB62115], [b:GB62125], [b:GB62135] : Touch panel in operation flag
// [b:GB62106], [b:GB62116], [b:GB62126], [b:GB62136] : Operation-end start trigger (start trigger of this
script)
// [b:GB62147], [b:GB62157], [b:GB62167], [b:GB62177] : the operation-end in process flag
// [u32:GD62002], [u32:GD62102], [u32:GD62202], [u32:GD62302] : Backup device for NET-IN0-15
// [u32:GD62034], [u32:GD62134], [u32:GD62234], [u32:GD62334] : Backup device for IN0-7
// [s32:GD61650], [s32:GD61750], [s32:GD61850], [s32:GD61950] : Backup device for Command position of
a monitor
// [s32:GD62052], [s32:GD62152], [s32:GD62252], [s32:GD62352] : Backup device for preset position
// [u32:GD62054],[u32:GD62154],[u32:GD62254],[u32:GD62354] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD62056], [w:GD62156], [w:GD62256], [w:GD62356] : Configuration execution waiting timer

// 9 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62106] == 1 && [b:GB62107] == 0){

  [b:GB62105] = 0;           // Clear touch panel in operation flag
  [b:GB62107] = 1;           // the operation-end in process flag (1: operation end in process)
  bmov([u32:GD62002],[1-9:u32:404449],16); // Restore NET-IN0-15 from backup
  bmov([u32:GD62034],[1-9:u32:404353],8);  // Restore IN0-7 from backup
  [1-9:u32:404169] = [u32:GD62054];       // Restore JOG moving distance from backup

  // Set the preset position parameter to the command position of the monitor

```

```

[s32:GD62050] = [1-9:s32:400199];
[1-9:s32:400909] = [s32:GD62050];

// Clear the input value of NET-IN
[1-9:w:400126] = 0;

// Execution of Configuration
[1-9:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD62056] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62056] > 1 && [b:GB62107] == 1){

    [1-9:s32:400395] = 1;           // Executes P-PRESET
    [1-9:s32:400909] = [s32:GD62052]; // Reset the Preset position parameter
    [1-9:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-9:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62106] = 0;               // Clear Operation-end start trigger
                                   // End the starting trigger in this script.
    [b:GB62107] = 0;               // Clear the operation-end in process flag
}

// 10 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62116] == 1 && [b:GB62117] == 0){

    [b:GB62115] = 0;               // Clear touch panel in operation flag
    [b:GB62117] = 1;               // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD62102],[1-10:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD62134],[1-10:u32:404353],8);  // Restore IN0-7 from backup
    [1-10:u32:404169] = [u32:GD62154];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62150] = [1-10:s32:400199];
    [1-10:s32:400909] = [s32:GD62150];

    // Clear the input value of NET-IN
    [1-10:w:400126] = 0;

    // Execution of Configuration
    [1-10:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62156] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62156] > 1 && [b:GB62117] == 1){

    [1-10:s32:400395] = 1;           // Executes P-PRESET
    [1-10:s32:400909] = [s32:GD62152]; // Reset the Preset position parameter
    [1-10:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-10:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62116] = 0;               // Clear Operation-end start trigger
                                   // End the starting trigger in this script.
    [b:GB62117] = 0;               // Clear the operation-end in process flag
}

// 11 axis

```

```

// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62126] == 1 && [b:GB62127] == 0){

    [b:GB62125] = 0;                // Clear touch panel in operation flag
    [b:GB62127] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD62202],[1-11:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD62234],[1-11:u32:404353],8);    // Restore IN0-7 from backup
    [1-11:u32:404169] = [u32:GD62254];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62250] = [1-11:s32:400199];
    [1-11:s32:400909] = [s32:GD62250];

    // Clear the input value of NET-IN
    [1-11:w:400126] = 0;

    // Execution of Configuration
    [1-11:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62256] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62256] > 1 && [b:GB62127] == 1){

    [1-11:s32:400395] = 1;                // Executes P-PRESET
    [1-11:s32:400909] = [s32:GD62252];    // Reset the Preset position parameter
    [1-11:s32:400397] = 0;                // Zero clear execution of Configuration
    [1-11:s32:400395] = 0;                // Zero clear for the execution of P-PRESET

    [b:GB62126] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62127] = 0;                // Clear the operation-end in process flag
}

// 12 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62136] == 1 && [b:GB62137] == 0){

    [b:GB62135] = 0;                // Clear touch panel in operation flag
    [b:GB62137] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD62302],[1-12:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD62334],[1-12:u32:404353],8);    // Restore IN0-7 from backup
    [1-12:u32:404169] = [u32:GD62354];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62350] = [1-12:s32:400199];
    [1-12:s32:400909] = [s32:GD62350];

    // Clear the input value of NET-IN
    [1-12:w:400126] = 0;

    // Execution of Configuration
    [1-12:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62356] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62356] > 1 && [b:GB62137] == 1){

    [1-12:s32:400395] = 1;                // Executes P-PRESET

```



[1-12:s32:400909] = [s32:GD62352]; // Reset the Preset position parameter [1-12:s32:400397] = 0; // Zero clear execution of Configuration [1-12:s32:400395] = 0; // Zero clear for the execution of P-PRESET  [b:GB62136] = 0; // Clear Operation-end start trigger // End the starting trigger in this script. [b:GB62137] = 0; // Clear the operation-end in process flag }			
Script No.	31024	Script Name	Script31024
Comment	9-12ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62110
Same as the script No. 31024 for the base screen 31024			
Script No.	31025	Script Name	Script31025
Comment	9-12_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62113
Same as the script No. 31025 for the base screen 31024			
Script No.	31026	Script Name	Script31026
Comment	9-12_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62115
Same as the script No. 31026 for the base screen 31024			
Script No.	31027	Script Name	Script31027
Comment	9-12_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62116
Same as the script No. 31027 for the base screen 31024			
Script No.	31024	Script Name	Script31024
Comment	9-12ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62120
Same as the script No. 31024 for the base screen 31024			
Script No.	31025	Script Name	Script31025
Comment	9-12_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62123
Same as the script No. 31025 for the base screen 31024			
Script No.	31026	Script Name	Script31026
Comment	9-12_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62125
Same as the script No. 31026 for the base screen 31024			
Script No.	31027	Script Name	Script31027
Comment	9-12_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62126
Same as the script No. 31027 for the base screen 31024			
Script No.	31024	Script Name	Script31024
Comment	9-12ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62130
Same as the script No. 31024 for the base screen 31024			
Script No.	31025	Script Name	Script31025
Comment	9-12_Multi ope prep		

Data Type	Signed BIN16	Trigger Type	ON GB62133
Same as the script No. 31025 for the base screen 31024			
Script No.	31026	Script Name	Script31026
Comment	9-12_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62135
Same as the script No. 31026 for the base screen 31024			
Script No.	31027	Script Name	Script31027
Comment	9-12_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62136
Same as the script No. 31027 for the base screen 31024			

### Base screen 31025

Script No.	31028	Script Name	Script31028
Comment	13-16ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62140
<pre>// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end // Screen script for 13-16 axis multiple operation  // 13-16_Check before start/end multiple operation script No.31028 // Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of 13-16 axis  // [b:GB62140] : 13 axis starting trigger using bit set // [b:GB62150] : 14 axis starting trigger using bit set // [b:GB62160] : 15 axis starting trigger using bit set // [b:GB62170] : 16 axis starting trigger using bit set // [b:GB62145] : 13 axis touch panel in operation flag // [b:GB62155] : 14 axis touch panel in operation flag // [b:GB62165] : 15 axis touch panel in operation flag // [b:GB62175] : 16 axis touch panel in operation flag  // [w:GD61000] = 1 means as below // Because Occurring alarm checking screen is responsible for station No. switching // Station No.(CPU No. of driver) is needed for alarm clear (Therefore settings of each axis is required.) // Screen switching device (GD61199) is responsible for Station No. reset  // 13 axis if([b:GB62140] == 1){     [w:GD61000] = 13;      // Alarm check     if([1-13:s32:400129] == 0){ // In case of no alarm         [b:GB62141] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)         [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.     } else { // in case the alarm occurring         [b:GB62141] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)         [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.     }      // In operation check     if([1-13:s32:400201] == 0){ // Case: not in operation         [b:GB62142] = 1; // In operation checking flag (1: not in operation)     } else { // In case the motor is running         [b:GB62142] = 0; // In operation checking flag (0: in operation)         [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.     } }</pre>			

```

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62141] == 1 && [b:GB62142] == 1){
    if([b:GB62145] == 0){ // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62140] = 0; // End the starting trigger of this script
        [b:GB62143] = 1; // Operation preparation start trigger
        [b:GB62141] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62142] = 0; // Clear the In operation checking flag
    } else { // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62140] = 0; // End the starting trigger of this script
        [b:GB62146] = 1; // Start trigger to terminate operation
        [b:GB62141] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62142] = 0; // Clear the In operation checking flag
    }
} else {
    [b:GB62140] = 0; // End the starting trigger of this script
    [b:GB62141] = 0; // Clear checking flag to see if an alarm is occurring
    [b:GB62142] = 0; // Clear the In operation checking flag
}
}

// 14 axis
if([b:GB62150] == 1){
    [w:GD61000] = 14;

    // Alarm check
    if([1-14:s32:400129] == 0){ // In case of no alarm
        [b:GB62151] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62151] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }
}

// In operation check
if([1-14:s32:400201] == 0){ // Case: not in operation
    [b:GB62152] = 1; // In operation checking flag (1: not in operation)
} else { // In case the motor is running
    [b:GB62152] = 0; // In operation checking flag (0: in operation)
    [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62151] == 1 && [b:GB62152] == 1){
    if([b:GB62155] == 0){ // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62150] = 0; // End the starting trigger of this script
        [b:GB62153] = 1; // Operation preparation start trigger
        [b:GB62151] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62152] = 0; // Clear the In operation checking flag
    } else { // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62150] = 0; // End the starting trigger of this script
        [b:GB62156] = 1; // Start trigger to terminate operation
        [b:GB62151] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62152] = 0; // Clear the In operation checking flag
    }
} else {
    [b:GB62150] = 0; // End the starting trigger of this script
    [b:GB62151] = 0; // Clear checking flag to see if an alarm is occurring
    [b:GB62152] = 0; // Clear the In operation checking flag
}
}

// 15 axis
if([b:GB62160] == 1){
    [w:GD61000] = 15;

```

```

// Alarm check
if([1-15:s32:400129] == 0){    // In case of no alarm
    [b:GB62161] = 1;          // Checking flag to see if an alarm is occurring(1: no alarm)
    [w:GD60004] = 0;          // Clear the overlap window for occurring alarm checking.
} else {                      // in case the alarm occurring
    [b:GB62161] = 0;          // Checking flag to see if an alarm is occurring(0: alarm occurring)
    [w:GD60004] = 32007;      // Display the overlap window for occurring alarm checking.
}

// In operation check
if([1-15:s32:400201] == 0){    // Case: not in operation
    [b:GB62162] = 1;          // In operation checking flag (1: not in operation)
} else {                      // In case the motor is running
    [b:GB62162] = 0;          // In operation checking flag (0: in operation)
    [w:GD60004] = 32008;      // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62161] == 1 && [b:GB62162] == 1){
    if([b:GB62165] == 0){      // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62160] = 0;      // End the starting trigger of this script
        [b:GB62163] = 1;      // Operation preparation start trigger
        [b:GB62161] = 0;      // Clear checking flag to see if an alarm is occurring
        [b:GB62162] = 0;      // Clear the In operation checking flag
    } else {                  // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62160] = 0;      // End the starting trigger of this script
        [b:GB62166] = 1;      // Start trigger to terminate operation
        [b:GB62161] = 0;      // Clear checking flag to see if an alarm is occurring
        [b:GB62162] = 0;      // Clear the In operation checking flag
    }
} else {
    [b:GB62160] = 0;          // End the starting trigger of this script
    [b:GB62161] = 0;          // Clear checking flag to see if an alarm is occurring
    [b:GB62162] = 0;          // Clear the In operation checking flag
}
}

// 16 axis
if([b:GB62170] == 1){
    [w:GD61000] = 16;

    // Alarm check
    if([1-16:s32:400129] == 0){    // In case of no alarm
        [b:GB62171] = 1;          // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;          // Clear the overlap window for occurring alarm checking.
    } else {                      // in case the alarm occurring
        [b:GB62171] = 0;          // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;      // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-16:s32:400201] == 0){    // Case: not in operation
        [b:GB62172] = 1;          // In operation checking flag (1: not in operation)
    } else {                      // In case the motor is running
        [b:GB62172] = 0;          // In operation checking flag (0: in operation))
        [w:GD60004] = 32008;      // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62171] == 1 && [b:GB62172] == 1){
        if([b:GB62175] == 0){      // Prepare Operation start in case the touch panel is not in operation. ゾ
            [b:GB62170] = 0;      // End the starting trigger of this script
            [b:GB62173] = 1;      // Operation preparation start trigger

```

```

        [b:GB62171] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62172] = 0;    // Clear the In operation checking flag
    } else {                // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62170] = 0;    // End the starting trigger of this script
        [b:GB62176] = 1;    // Start trigger to terminate operation
        [b:GB62171] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62172] = 0;    // Clear the In operation checking flag
    }
} else {
    [b:GB62170] = 0;    // End the starting trigger of this script
    [b:GB62171] = 0;    // Clear checking flag to see if an alarm is occurring
    [b:GB62172] = 0;    // Clear the In operation checking flag
}
}
}

```

Script No.	31029	Script Name	Script31029
Comment	13-16_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62143

```

// Script to prepare for touch panel operation
// Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter
// Screen script for 13-16 axis multiple operation

// 13-16_Multiple operation preparation script No.31029
// Start the script in case GB62143, GB62153, GB62163, GB62173 in the 13-16_Check before start/end
multiple operation script (No.31028) are ON.

// [b:GB62143], [b:GB62153], [b:GB62163], [b:GB62173] : Operation preparation start trigger (start trigger of
this script)
// [b:GB62144], [b:GB62154], [b:GB62164], [b:GB62174] : Preparing operation flag
// [b:GB62145], [b:GB62155], [b:GB62165], [b:GB62175] : Touch panel in operation flag
// [u32:GD62402], [u32:GD62502], [u32:GD62602], [u32:GD62702] : Backup device for NET-IN0-15
// [u32:GD62434], [u32:GD62534], [u32:GD62634], [u32:GD62734] : Backup device for IN0-7
// [s32:GD62450], [s32:GD62550], [s32:GD62650], [s32:GD62750] : Backup device for Command position of
a monitor
// [s32:GD62452], [s32:GD62552], [s32:GD62652], [u32:GD62752] : Backup device for preset position
// [u32:GD62454],[u32:GD62554],[u32:GD62654],[u32:GD62754] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD62456], [w:GD62556], [w:GD62656], [w:GD62756] : Configuration execution waiting timer

// 13 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62143] == 1 && [b:GB62144] == 0){

    [b:GB62144] = 1;                // Preparing operation flag (1: Preparing operation)
    bmov([1-13:u32:404449],[u32:GD62402],16);    // Backs up NET-IN0-15
    bmov([1-13:u32:404353],[u32:GD62434],8);    // Backs up IN0-7
    [s32:GD62452] = [1-13:s32:400909];    // Backs up the preset position
    [u32:GD62454] = [1-13:u32:404169];    // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-13:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-13:u32:404449] = 48;    // NET-IN0  M0
    [1-13:u32:404451] = 49;    // NET-IN1  M1
    [1-13:u32:404453] = 50;    // NET-IN2  M2
    [1-13:u32:404455] = 51;    // NET-IN3  M3
    [1-13:u32:404457] = 52;    // NET-IN4  M4
    [1-13:u32:404459] = 53;    // NET-IN5  M5
    [1-13:u32:404461] = 4;    // NET-IN6  START
    [1-13:u32:404463] = 3;    // NET-IN7  HOME
    [1-13:u32:404465] = 18;    // NET-IN8  STOP
    [1-13:u32:404467] = 9;    // NET-IN9  MS1
    [1-13:u32:404469] = 10;    // NET-IN10 MS2

```

```

[1-13:u32:404471] = 5;    // NET-IN11 SSTART
[1-13:u32:404473] = 6;    // NET-IN12 +JOG
[1-13:u32:404475] = 7;    // NET-IN13 -JOG
[1-13:u32:404477] = 1;    // NET-IN14 FWD
[1-13:u32:404479] = 2;    // NET-IN15 RVS

[1-13:u32:404353] = 32; // IN0 R0
[1-13:u32:404355] = 33; // IN1 R1
[1-13:u32:404357] = 34; // IN2 R2
[1-13:u32:404359] = 35; // IN3 R3
[1-13:u32:404361] = 36; // IN4 R4
[1-13:u32:404363] = 37; // IN5 R5
[1-13:u32:404365] = 18; // IN6 STOP
[1-13:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD62450] = [1-13:s32:400199];
[1-13:s32:400909] = [s32:GD62450];

// Execution of Configuration
[1-13:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD62456] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62456] > 1 && [b:GB62144] == 1){

    [1-13:s32:400395] = 1;    // Executes P-PRESET
    [1-13:s32:400909] = [s32:GD62452]; // Restore the preset position parameter after backup
    [1-13:s32:400397] = 0;    // Zero clear execution of Configuration
    [1-13:s32:400395] = 0;    // Zero clear for the execution of P-PRESET

    [b:GB62143] = 0;    // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62144] = 0;    // Clear Preparing operation flag
    [b:GB62145] = 1;    // Touch panel in operation flag (1: in operation)
}

// 14 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62153] == 1 && [b:GB62154] == 0){

    [b:GB62154] = 1;    // Preparing operation flag (1: Preparing operation)
    bmov([1-14:u32:404449],[u32:GD62502],16); // Backs up NET-IN0-15
    bmov([1-14:u32:404353],[u32:GD62534],8); // Backs up IN0-7
    [s32:GD62552] = [1-14:s32:400909]; // Backs up the preset position
    [u32:GD62554] = [1-14:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-14:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-14:u32:404449] = 48; // NET-IN0 M0
    [1-14:u32:404451] = 49; // NET-IN1 M1
    [1-14:u32:404453] = 50; // NET-IN2 M2
    [1-14:u32:404455] = 51; // NET-IN3 M3
    [1-14:u32:404457] = 52; // NET-IN4 M4
    [1-14:u32:404459] = 53; // NET-IN5 M5
    [1-14:u32:404461] = 4; // NET-IN6 START
    [1-14:u32:404463] = 3; // NET-IN7 HOME
    [1-14:u32:404465] = 18; // NET-IN8 STOP

```

```

[1-14:u32:404467] = 9;    // NET-IN9  MS1
[1-14:u32:404469] = 10;   // NET-IN10 MS2
[1-14:u32:404471] = 5;    // NET-IN11 SSTART
[1-14:u32:404473] = 6;    // NET-IN12 +JOG
[1-14:u32:404475] = 7;    // NET-IN13 -JOG
[1-14:u32:404477] = 1;    // NET-IN14 FWD
[1-14:u32:404479] = 2;    // NET-IN15 RVS

[1-14:u32:404353] = 32; // IN0 R0
[1-14:u32:404355] = 33; // IN1 R1
[1-14:u32:404357] = 34; // IN2 R2
[1-14:u32:404359] = 35; // IN3 R3
[1-14:u32:404361] = 36; // IN4 R4
[1-14:u32:404363] = 37; // IN5 R5
[1-14:u32:404365] = 18; // IN6 STOP
[1-14:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD62550] = [1-14:s32:400199];
[1-14:s32:400909] = [s32:GD62550];

// Execution of Configuration
[1-14:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD62556] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62556] > 1 && [b:GB62154] == 1){

    [1-14:s32:400395] = 1;           // Executes P-PRESET
    [1-14:s32:400909] = [s32:GD62552]; // Restore the preset position parameter after backup
    [1-14:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-14:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62153] = 0;                // Clear Operation preparation start trigger
                                    // End the starting trigger in this script.
    [b:GB62154] = 0;                // Clear Preparing operation flag
    [b:GB62155] = 1;                // Touch panel in operation flag (1: in operation)
}

// 15 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62163] == 1 && [b:GB62164] == 0){

    [b:GB62164] = 1;                // Preparing operation flag (1: Preparing operation)
    bmov([1-15:u32:404449],[u32:GD62602],16); // Backs up NET-IN0-15
    bmov([1-15:u32:404353],[u32:GD62634],8);  // Backs up IN0-7
    [s32:GD62652] = [1-15:s32:400909];        // Backs up the preset position
    [u32:GD62654] = [1-15:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-15:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-15:u32:404449] = 48; // NET-IN0  M0
    [1-15:u32:404451] = 49; // NET-IN1  M1
    [1-15:u32:404453] = 50; // NET-IN2  M2
    [1-15:u32:404455] = 51; // NET-IN3  M3
    [1-15:u32:404457] = 52; // NET-IN4  M4
    [1-15:u32:404459] = 53; // NET-IN5  M5
    [1-15:u32:404461] = 4;  // NET-IN6  START

```

```

[1-15:u32:404463] = 3;    // NET-IN7  HOME
[1-15:u32:404465] = 18;   // NET-IN8  STOP
[1-15:u32:404467] = 9;    // NET-IN9  MS1
[1-15:u32:404469] = 10;   // NET-IN10 MS2
[1-15:u32:404471] = 5;    // NET-IN11 SSTART
[1-15:u32:404473] = 6;    // NET-IN12 +JOG
[1-15:u32:404475] = 7;    // NET-IN13 -JOG
[1-15:u32:404477] = 1;    // NET-IN14 FWD
[1-15:u32:404479] = 2;    // NET-IN15 RVS

[1-15:u32:404353] = 32; // IN0 R0
[1-15:u32:404355] = 33; // IN1 R1
[1-15:u32:404357] = 34; // IN2 R2
[1-15:u32:404359] = 35; // IN3 R3
[1-15:u32:404361] = 36; // IN4 R4
[1-15:u32:404363] = 37; // IN5 R5
[1-15:u32:404365] = 18; // IN6 STOP
[1-15:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD62650] = [1-15:s32:400199];
[1-15:s32:400909] = [s32:GD62650];

// Execution of Configuration
[1-15:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD62656] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62656] > 1 && [b:GB62164] == 1){

    [1-15:s32:400395] = 1;    // Executes P-PRESET
    [1-15:s32:400909] = [s32:GD62652]; // Restore the preset position parameter after backup
    [1-15:s32:400397] = 0;    // Zero clear execution of Configuration
    [1-15:s32:400395] = 0;    // Zero clear for the execution of P-PRESET

    [b:GB62163] = 0;    // Clear Operation preparation start trigger
                        // End the starting trigger in this script.
    [b:GB62164] = 0;    // Clear Preparing operation flag
    [b:GB62165] = 1;    // Touch panel in operation flag (1: in operation)
}

// 16 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62173] == 1 && [b:GB62174] == 0){

    [b:GB62174] = 1;    // Preparing operation flag (1: Preparing operation)
    bmov([1-16:u32:404449],[u32:GD62702],16); // Backs up NET-IN0-15
    bmov([1-16:u32:404353],[u32:GD62734],8); // Backs up IN0-7
    [s32:GD62752] = [1-16:s32:400909]; // Backs up the preset position
    [u32:GD62754] = [1-16:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-16:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-16:u32:404449] = 48; // NET-IN0  M0
    [1-16:u32:404451] = 49; // NET-IN1  M1
    [1-16:u32:404453] = 50; // NET-IN2  M2
    [1-16:u32:404455] = 51; // NET-IN3  M3
    [1-16:u32:404457] = 52; // NET-IN4  M4

```



```

[1-16:u32:404459] = 53; // NET-IN5 M5
[1-16:u32:404461] = 4; // NET-IN6 START
[1-16:u32:404463] = 3; // NET-IN7 HOME
[1-16:u32:404465] = 18; // NET-IN8 STOP
[1-16:u32:404467] = 9; // NET-IN9 MS1
[1-16:u32:404469] = 10; // NET-IN10 MS2
[1-16:u32:404471] = 5; // NET-IN11 SSTART
[1-16:u32:404473] = 6; // NET-IN12 +JOG
[1-16:u32:404475] = 7; // NET-IN13 -JOG
[1-16:u32:404477] = 1; // NET-IN14 FWD
[1-16:u32:404479] = 2; // NET-IN15 RVS

```

```

[1-16:u32:404353] = 32; // IN0 R0
[1-16:u32:404355] = 33; // IN1 R1
[1-16:u32:404357] = 34; // IN2 R2
[1-16:u32:404359] = 35; // IN3 R3
[1-16:u32:404361] = 36; // IN4 R4
[1-16:u32:404363] = 37; // IN5 R5
[1-16:u32:404365] = 18; // IN6 STOP
[1-16:u32:404367] = 39; // IN7 R7

```

```

// Set the preset position parameter to the command position of the monitor
[s32:GD62750] = [1-16:s32:400199];
[1-16:s32:400909] = [s32:GD62750];

```

```

// Execution of Configuration
[1-16:s32:400397] = 1;

```

```

// Start recording of Configuration execution waiting timer [sec]
[w:GD62756] = [w:GS7];

```

```

}

```

```

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62756] > 1 && [b:GB62174] == 1){

```

```

[1-16:s32:400395] = 1; // Executes P-PRESET
[1-16:s32:400909] = [s32:GD62752]; // Restore the preset position parameter after backup
[1-16:s32:400397] = 0; // Zero clear execution of Configuration
[1-16:s32:400395] = 0; // Zero clear for the execution of P-PRESET

```

```

[b:GB62173] = 0; // Clear Operation preparation start trigger
// End the starting trigger in this script.
[b:GB62174] = 0; // Clear Preparing operation flag
[b:GB62175] = 1; // Touch panel in operation flag (1: in operation)
}

```

Script No.	31030	Script Name	Script31030
Comment	13-16_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62145

```

// Script for positioning operation with touch panel operation
// Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No.
// After being changed, M0-M5 are turned ON by driver input command (400126).

// 13-16_Script in multiple operation No.31030
// Start the script in case GB62145, GB62155, GB62165, GB62175 in the 13-16_multiple operation
preparation script (No.31029) are ON.

// [b:GB62145],[b:GB62155],[b:GB62165],[b:GB62175] : Touch panel in operation flag(start trigger of this
script)
// Bit also works as interlock not to change the screen while touch panel in operation

// [1-13:u32:400195] - [1-16:u32:400195] : Modbus address of selected data No.
// [u32:GD62400], [u32:GD62500], [GD62600], [u32:GD62700] : Device for the operation data No.

// Change M0-M5 of driver input command to reflect the selected operation data No.

```

```

// To ease the touch panel processing load,
// In case the currently selected data No. (400915) of the monitor and the selected operation data No. are not
the same,
// change the driver input command (400126) to reflect a change

// 13 axis
if([b:GB62145] == 1){

    if([1-13:u32:400195] != [u32:GD62400]){
        [1-13:b:400126.b0] = [b:GD62400.b0]; // M0
        [1-13:b:400126.b1] = [b:GD62400.b1]; // M1
        [1-13:b:400126.b2] = [b:GD62400.b2]; // M2
        [1-13:b:400126.b3] = [b:GD62400.b3]; // M3
        [1-13:b:400126.b4] = [b:GD62400.b4]; // M4
        [1-13:b:400126.b5] = [b:GD62400.b5]; // M5
    }
}

// 14 axis
if([b:GB62155] == 1){

    if([1-14:u32:400195] != [u32:GD62500]){
        [1-14:b:400126.b0] = [b:GD62500.b0]; // M0
        [1-14:b:400126.b1] = [b:GD62500.b1]; // M1
        [1-14:b:400126.b2] = [b:GD62500.b2]; // M2
        [1-14:b:400126.b3] = [b:GD62500.b3]; // M3
        [1-14:b:400126.b4] = [b:GD62500.b4]; // M4
        [1-14:b:400126.b5] = [b:GD62500.b5]; // M5
    }
}

// 15 axis
if([b:GB62165] == 1){

    if([1-15:u32:400195] != [u32:GD62600]){
        [1-15:b:400126.b0] = [b:GD62600.b0]; // M0
        [1-15:b:400126.b1] = [b:GD62600.b1]; // M1
        [1-15:b:400126.b2] = [b:GD62600.b2]; // M2
        [1-15:b:400126.b3] = [b:GD62600.b3]; // M3
        [1-15:b:400126.b4] = [b:GD62600.b4]; // M4
        [1-15:b:400126.b5] = [b:GD62600.b5]; // M5
    }
}

// 16 axis
if([b:GB62175] == 1){

    if([1-16:u32:400195] != [u32:GD62700]){
        [1-16:b:400126.b0] = [b:GD62700.b0]; // M0
        [1-16:b:400126.b1] = [b:GD62700.b1]; // M1
        [1-16:b:400126.b2] = [b:GD62700.b2]; // M2
        [1-16:b:400126.b3] = [b:GD62700.b3]; // M3
        [1-16:b:400126.b4] = [b:GD62700.b4]; // M4
        [1-16:b:400126.b5] = [b:GD62700.b5]; // M5
    }
}

```

Script No.	31031	Script Name	Script31031
Comment	13-16_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62146
// Script to disable touch panel operation // Restore such as the NET-IN signal and the necessary parameter from backup // Screen script for 13-16 axis multiple operation  // 13-16_End multiple operation script No.31031			

```

// Start the script in case GB62146, GB62156, GB62166, GB62176 in the 13-16_Check before start/end
multiple operation script (No.31028) are ON.

// [b:GB62145], [b:GB62155], [b:GB62165], [b:GB62175] : Touch panel in operation flag
// [b:GB62146], [b:GB62156], [b:GB62166], [b:GB62176] : Operation-end start trigger (start trigger of this
script)
// [b:GB62147], [b:GB62157], [b:GB62167], [b:GB62177] : the operation-end in process flag
// [u32:GD62402], [u32:GD62502], [u32:GD62602], [u32:GD62702] : Backup device for NET-IN0-15
// [u32:GD62434], [u32:GD62534], [u32:GD62634], [u32:GD62734] : Backup device for IN0-7
// [s32:GD62450], [s32:GD62550], [s32:GD62650], [s32:GD62750] : Backup device for Command position of
a monitor
// [s32:GD62452], [s32:GD62552], [s32:GD62652], [s32:GD62752] : Backup device for preset position
// [u32:GD62454],[u32:GD62554],[u32:GD62654],[u32:GD62754] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD62456], [w:GD62556], [w:GD62656], [w:GD62756] : Configuration execution waiting timer

// 13 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62146] == 1 && [b:GB62147] == 0){

    [b:GB62145] = 0;                // Clear touch panel in operation flag
    [b:GB62147] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD62402],[1-13:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD62434],[1-13:u32:404353],8);    // Restore IN0-7 from backup
    [1-13:u32:404169] = [u32:GD62454];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62450] = [1-13:s32:400199];
    [1-13:s32:400909] = [s32:GD62450];

    // Clear the input value of NET-IN
    [1-13:w:400126] = 0;

    // Execution of Configuration
    [1-13:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62456] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62456] > 1 && [b:GB62147] == 1){

    [1-13:s32:400395] = 1;                // Executes P-PRESET
    [1-13:s32:400909] = [s32:GD62452];    // Reset the Preset position parameter
    [1-13:s32:400397] = 0;                // Zero clear execution of Configuration
    [1-13:s32:400395] = 0;                // Zero clear for the execution of P-PRESET

    [b:GB62146] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62147] = 0;                // Clear the operation-end in process flag
}

// 14 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62156] == 1 && [b:GB62157] == 0){

    [b:GB62155] = 0;                // Clear touch panel in operation flag
    [b:GB62157] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD62502],[1-14:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD62534],[1-14:u32:404353],8);    // Restore IN0-7 from backup
    [1-14:u32:404169] = [u32:GD62554];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor

```

```

[s32:GD62550] = [1-14:s32:400199];
[1-14:s32:400909] = [s32:GD62550];

// Clear the input value of NET-IN
[1-14:w:400126] = 0;

// Execution of Configuration
[1-14:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD62556] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62556] > 1 && [b:GB62157] == 1){

    [1-14:s32:400395] = 1;           // Executes P-PRESET
    [1-14:s32:400909] = [s32:GD62552]; // Reset the Preset position parameter
    [1-14:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-14:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62156] = 0;                // Clear Operation-end start trigger
                                    // End the starting trigger in this script.
    [b:GB62157] = 0;                // Clear the operation-end in process flag
}

// 15 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62166] == 1 && [b:GB62167] == 0){

    [b:GB62165] = 0;                // Clear touch panel in operation flag
    [b:GB62167] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD62602],[1-15:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD62634],[1-15:u32:404353],8);  // Restore IN0-7 from backup
    [1-15:u32:404169] = [u32:GD62654];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62650] = [1-15:s32:400199];
    [1-15:s32:400909] = [s32:GD62650];

    // Clear the input value of NET-IN
    [1-15:w:400126] = 0;

    // Execution of Configuration
    [1-15:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62656] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62656] > 1 && [b:GB62167] == 1){

    [1-15:s32:400395] = 1;           // Executes P-PRESET
    [1-15:s32:400909] = [s32:GD62652]; // Reset the Preset position parameter
    [1-15:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-15:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62166] = 0;                // Clear Operation-end start trigger
                                    // End the starting trigger in this script.
    [b:GB62167] = 0;                // Clear the operation-end in process flag
}

// 16 axis

```

```

// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62176] == 1 && [b:GB62177] == 0){

    [b:GB62175] = 0;                // Clear touch panel in operation flag
    [b:GB62177] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD62702],[1-16:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD62734],[1-16:u32:404353],8);    // Restore IN0-7 from backup
    [1-16:u32:404169] = [u32:GD62754];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62750] = [1-16:s32:400199];
    [1-16:s32:400909] = [s32:GD62750];

    // Clear the input value of NET-IN
    [1-16:w:400126] = 0;

    // Execution of Configuration
    [1-16:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62756] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62756] > 1 && [b:GB62177] == 1){

    [1-16:s32:400395] = 1;                // Executes P-PRESET
    [1-16:s32:400909] = [s32:GD62752];    // Reset the Preset position parameter
    [1-16:s32:400397] = 0;                // Zero clear execution of Configuration
    [1-16:s32:400395] = 0;                // Zero clear for the execution of P-PRESET

    [b:GB62176] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62177] = 0;                // Clear the operation-end in process flag
}

```

Script No.	31028	Script Name	Script31028
Comment	13-16ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62150
Same as the script No. 31028 for the base screen 31025			
Script No.	31029	Script Name	Script31029
Comment	13-16_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62153
Same as the script No. 31029 for the base screen 31025			
Script No.	31030	Script Name	Script31030
Comment	13-16_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62155
Same as the script No. 31030 for the base screen 31025			
Script No.	31031	Script Name	Script31031
Comment	13-16_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62156
Same as the script No. 31031 for the base screen 31025			
Script No.	31028	Script Name	Script31028
Comment	13-16ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62160

Same as the script No. 31028 for the base screen 31025			
Script No.	31029	Script Name	Script31029
Comment	13-16_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62163
Same as the script No. 31029 for the base screen 31025			
Script No.	31030	Script Name	Script31030
Comment	13-16_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62165
Same as the script No. 31030 for the base screen 31025			
Script No.	31031	Script Name	Script31031
Comment	13-16_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62166
Same as the script No. 31031 for the base screen 31025			
Script No.	31028	Script Name	Script31028
Comment	13-16ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62170
Same as the script No. 31028 for the base screen 31025			
Script No.	31029	Script Name	Script31029
Comment	13-16_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62173
Same as the script No. 31029 for the base screen 31025			
Script No.	31030	Script Name	Script31030
Comment	13-16_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62175
Same as the script No. 31030 for the base screen 31025			
Script No.	31031	Script Name	Script31031
Comment	13-16_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62176
Same as the script No. 31031 for the base screen 31025			

#### Base screen 31026

Script No.	31032	Script Name	Script31032
Comment	17-20ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62180
// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end // Screen script for 17-20 axis multiple operation  // 17-20_Check before start/end multiple operation script No.31032 // Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of 17-20 axis  // [b:GB62180] : 17 axis starting trigger using bit set // [b:GB62190] : 18 axis starting trigger using bit set // [b:GB62200] : 19 axis starting trigger using bit set // [b:GB62210] : 20 axis starting trigger using bit set // [b:GB62185] : 17 axis touch panel in operation flag // [b:GB62195] : 18 axis touch panel in operation flag			

```

// [b:GB62205] : 19 axis touch panel in operation flag
// [b:GB62215] : 20 axis touch panel in operation flag

// [w:GD61000] = 1 means as below
// Because Occurring alarm checking screen is responsible for station No. switching
// Station No.(CPU No. of driver) is needed for alarm clear (Therefore settings of each axis is required.)
// Screen switching device (GD61199) is responsible for Station No. reset

// 17 axis
if([b:GB62180] == 1){
    [w:GD61000] = 17;

    // Alarm check
    if([1-17:s32:400129] == 0){ // In case of no alarm
        [b:GB62181] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62181] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-17:s32:400201] == 0){ // Case: not in operation
        [b:GB62182] = 1; // In operation checking flag (1: not in operation)
    } else { // In case the motor is running
        [b:GB62182] = 0; // In operation checking flag (0: in operation)
        [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62181] == 1 && [b:GB62182] == 1){
        if([b:GB62185] == 0){ // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62180] = 0; // End the starting trigger of this script
            [b:GB62183] = 1; // Operation preparation start trigger
            [b:GB62181] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62182] = 0; // Clear the In operation checking flag
        } else { // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62180] = 0; // End the starting trigger of this script
            [b:GB62186] = 1; // Start trigger to terminate operation
            [b:GB62181] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62182] = 0; // Clear the In operation checking flag
        }
    } else {
        [b:GB62180] = 0; // End the starting trigger of this script
        [b:GB62181] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62182] = 0; // Clear the In operation checking flag
    }
}

// 18 axis
if([b:GB62190] == 1){
    [w:GD61000] = 18;

    // Alarm check
    if([1-18:s32:400129] == 0){ // In case of no alarm
        [b:GB62191] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62191] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-18:s32:400201] == 0){ // Case: not in operation

```

```

    [b:GB62192] = 1;          // In operation checking flag (1: not in operation)
} else {                    // In case the motor is running
    [b:GB62192] = 0;          // In operation checking flag (0: in operation)
    [w:GD60004] = 32008;      // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62191] == 1 && [b:GB62192] == 1){
    if([b:GB62195] == 0){    // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62190] = 0;    // End the starting trigger of this script
        [b:GB62193] = 1;    // Operation preparation start trigger
        [b:GB62191] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62192] = 0;    // Clear the In operation checking flag
    } else {                // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62190] = 0;    // End the starting trigger of this script
        [b:GB62196] = 1;    // Start trigger to terminate operation
        [b:GB62191] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62192] = 0;    // Clear the In operation checking flag
    }
} else {
    [b:GB62190] = 0;        // End the starting trigger of this script
    [b:GB62191] = 0;        // Clear checking flag to see if an alarm is occurring
    [b:GB62192] = 0;        // Clear the In operation checking flag
}
}

// 19 axis
if([b:GB62200] == 1){
    [w:GD61000] = 19;

    // Alarm check
    if([1-19:s32:400129] == 0){    // In case of no alarm
        [b:GB62201] = 1;          // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;          // Clear the overlap window for occurring alarm checking.
    } else {                      // in case the alarm occurring
        [b:GB62201] = 0;          // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;      // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-19:s32:400201] == 0){    // Case: not in operation
        [b:GB62202] = 1;          // In operation checking flag (1: not in operation)
    } else {                      // In case the motor is running
        [b:GB62202] = 0;          // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;      // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62201] == 1 && [b:GB62202] == 1){
        if([b:GB62205] == 0){    // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62200] = 0;    // End the starting trigger of this script
            [b:GB62203] = 1;    // Operation preparation start trigger
            [b:GB62201] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62202] = 0;    // Clear the In operation checking flag
        } else {                // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62200] = 0;    // End the starting trigger of this script
            [b:GB62206] = 1;    // Start trigger to terminate operation
            [b:GB62201] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62202] = 0;    // Clear the In operation checking flag
        }
    } else {
        [b:GB62200] = 0;        // End the starting trigger of this script
        [b:GB62201] = 0;        // Clear checking flag to see if an alarm is occurring
        [b:GB62202] = 0;        // Clear the In operation checking flag
    }
}

```



```

}
}

// 20 axis
if([b:GB62210] == 1){
    [w:GD61000] = 20;

    // Alarm check
    if([1-20:s32:400129] == 0){ // In case of no alarm
        [b:GB62211] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62211] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-20:s32:400201] == 0){ // Case: not in operation
        [b:GB62212] = 1; // In operation checking flag (1: not in operation)
    } else { // In case the motor is running
        [b:GB62212] = 0; // In operation checking flag (0: in operation)
        [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62211] == 1 && [b:GB62212] == 1){
        if([b:GB62215] == 0){ // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62210] = 0; // End the starting trigger of this script
            [b:GB62213] = 1; // Operation preparation start trigger
            [b:GB62211] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62212] = 0; // Clear the In operation checking flag
        } else { // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62210] = 0; // End the starting trigger of this script
            [b:GB62216] = 1; // Start trigger to terminate operation
            [b:GB62211] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62212] = 0; // Clear the In operation checking flag
        }
    } else {
        [b:GB62210] = 0; // End the starting trigger of this script
        [b:GB62211] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62212] = 0; // Clear the In operation checking flag
    }
}
}

```

Script No.	31033	Script Name	Script31033
Comment	17-20_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62183

```

// Script to prepare for touch panel operation
// Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter
// Screen script for 17-20 axis multiple operation

// 17-20_Multiple operation preparation script No.31033
// Start the script in case GB62183, GB62193, GB62203, GB62213 in the 17-20_Check before start/end
multiple operation script (No.31032) are ON.

// [b:GB62183], [b:GB62193], [b:GB62203], [b:GB62213] : Operation preparation start trigger (start trigger of
this script)
// [b:GB62184], [b:GB62194], [b:GB62204], [b:GB62214] : Preparing operation flag
// [b:GB62185], [b:GB62195], [b:GB62205], [b:GB62215] : Touch panel in operation flag
// [u32:GD62802], [u32:GD62902], [u32:GD63002], [u32:GD63102] : Backup device for NET-IN0-15
// [u32:GD62834], [u32:GD62934], [u32:GD63034], [u32:GD63134] : Backup device for IN0-7
// [s32:GD62850], [s32:GD62950], [s32:GD63050], [s32:GD63150] : Backup device for Command position of
a monitor
// [s32:GD62852], [s32:GD62952], [s32:GD63052], [u32:GD63152] : Backup device for preset position
// [u32:GD62854],[u32:GD62954],[u32:GD63054],[u32:GD63154] : Backup device for JOG moving distance

```

```

(minimum moving distance for GOT)
// [w:GD62856], [w:GD62956], [w:GD63056], [w:GD63156] : Configuration execution waiting timer

// 17 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62183] == 1 && [b:GB62184] == 0){

    [b:GB62184] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-17:u32:404449],[u32:GD62802],16); // Backs up NET-IN0-15
    bmov([1-17:u32:404353],[u32:GD62834],8); // Backs up IN0-7
    [s32:GD62852] = [1-17:s32:400909]; // Backs up the preset position
    [u32:GD62854] = [1-17:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-17:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-17:u32:404449] = 48; // NET-IN0 M0
    [1-17:u32:404451] = 49; // NET-IN1 M1
    [1-17:u32:404453] = 50; // NET-IN2 M2
    [1-17:u32:404455] = 51; // NET-IN3 M3
    [1-17:u32:404457] = 52; // NET-IN4 M4
    [1-17:u32:404459] = 53; // NET-IN5 M5
    [1-17:u32:404461] = 4; // NET-IN6 START
    [1-17:u32:404463] = 3; // NET-IN7 HOME
    [1-17:u32:404465] = 18; // NET-IN8 STOP
    [1-17:u32:404467] = 9; // NET-IN9 MS1
    [1-17:u32:404469] = 10; // NET-IN10 MS2
    [1-17:u32:404471] = 5; // NET-IN11 SSTART
    [1-17:u32:404473] = 6; // NET-IN12 +JOG
    [1-17:u32:404475] = 7; // NET-IN13 -JOG
    [1-17:u32:404477] = 1; // NET-IN14 FWD
    [1-17:u32:404479] = 2; // NET-IN15 RVS

    [1-17:u32:404353] = 32; // IN0 R0
    [1-17:u32:404355] = 33; // IN1 R1
    [1-17:u32:404357] = 34; // IN2 R2
    [1-17:u32:404359] = 35; // IN3 R3
    [1-17:u32:404361] = 36; // IN4 R4
    [1-17:u32:404363] = 37; // IN5 R5
    [1-17:u32:404365] = 18; // IN6 STOP
    [1-17:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62850] = [1-17:s32:400199];
    [1-17:s32:400909] = [s32:GD62850];

    // Execution of Configuration
    [1-17:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62856] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62856] > 1 && [b:GB62184] == 1){

    [1-17:s32:400395] = 1; // Executes P-PRESET
    [1-17:s32:400909] = [s32:GD62852]; // Restore the preset position parameter after backup
    [1-17:s32:400397] = 0; // Zero clear execution of Configuration
    [1-17:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62183] = 0; // Clear Operation preparation start trigger

```

```

// End the starting trigger in this script.
[b:GB62184] = 0;           // Clear Preparing operation flag
[b:GB62185] = 1;           // Touch panel in operation flag (1: in operation)
}

// 18 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62193] == 1 && [b:GB62194] == 0){

    [b:GB62194] = 1;           // Preparing operation flag (1: Preparing operation)
    bmov([1-18:u32:404449],[u32:GD62902],16); // Backs up NET-IN0-15
    bmov([1-18:u32:404353],[u32:GD62934],8); // Backs up IN0-7
    [s32:GD62952] = [1-18:s32:400909]; // Backs up the preset position
    [u32:GD62954] = [1-18:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-18:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-18:u32:404449] = 48; // NET-IN0 M0
    [1-18:u32:404451] = 49; // NET-IN1 M1
    [1-18:u32:404453] = 50; // NET-IN2 M2
    [1-18:u32:404455] = 51; // NET-IN3 M3
    [1-18:u32:404457] = 52; // NET-IN4 M4
    [1-18:u32:404459] = 53; // NET-IN5 M5
    [1-18:u32:404461] = 4; // NET-IN6 START
    [1-18:u32:404463] = 3; // NET-IN7 HOME
    [1-18:u32:404465] = 18; // NET-IN8 STOP
    [1-18:u32:404467] = 9; // NET-IN9 MS1
    [1-18:u32:404469] = 10; // NET-IN10 MS2
    [1-18:u32:404471] = 5; // NET-IN11 SSTART
    [1-18:u32:404473] = 6; // NET-IN12 +JOG
    [1-18:u32:404475] = 7; // NET-IN13 -JOG
    [1-18:u32:404477] = 1; // NET-IN14 FWD
    [1-18:u32:404479] = 2; // NET-IN15 RVS

    [1-18:u32:404353] = 32; // IN0 R0
    [1-18:u32:404355] = 33; // IN1 R1
    [1-18:u32:404357] = 34; // IN2 R2
    [1-18:u32:404359] = 35; // IN3 R3
    [1-18:u32:404361] = 36; // IN4 R4
    [1-18:u32:404363] = 37; // IN5 R5
    [1-18:u32:404365] = 18; // IN6 STOP
    [1-18:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62950] = [1-18:s32:400199];
    [1-18:s32:400909] = [s32:GD62950];

    // Execution of Configuration
    [1-18:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62956] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD62956] > 1 && [b:GB62194] == 1){

    [1-18:s32:400395] = 1; // Executes P-PRESET
    [1-18:s32:400909] = [s32:GD62952]; // Restore the preset position parameter after backup
    [1-18:s32:400397] = 0; // Zero clear execution of Configuration
    [1-18:s32:400395] = 0; // Zero clear for the execution of P-PRESET
}

```

```

[b:GB62193] = 0;          // Clear Operation preparation start trigger
                        // End the starting trigger in this script.
[b:GB62194] = 0;          // Clear Preparing operation flag
[b:GB62195] = 1;          // Touch panel in operation flag (1: in operation)
}

// 19 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62203] == 1 && [b:GB62204] == 0){

    [b:GB62204] = 1;      // Preparing operation flag (1: Preparing operation)
    bmov([1-19:u32:404449],[u32:GD63002],16); // Backs up NET-IN0-15
    bmov([1-19:u32:404353],[u32:GD63034],8);  // Backs up IN0-7
    [s32:GD63052] = [1-19:s32:400909];        // Backs up the preset position
    [u32:GD63054] = [1-19:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-19:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-19:u32:404449] = 48; // NET-IN0  M0
    [1-19:u32:404451] = 49; // NET-IN1  M1
    [1-19:u32:404453] = 50; // NET-IN2  M2
    [1-19:u32:404455] = 51; // NET-IN3  M3
    [1-19:u32:404457] = 52; // NET-IN4  M4
    [1-19:u32:404459] = 53; // NET-IN5  M5
    [1-19:u32:404461] = 4;  // NET-IN6  START
    [1-19:u32:404463] = 3;  // NET-IN7  HOME
    [1-19:u32:404465] = 18; // NET-IN8  STOP
    [1-19:u32:404467] = 9;  // NET-IN9  MS1
    [1-19:u32:404469] = 10; // NET-IN10 MS2
    [1-19:u32:404471] = 5;  // NET-IN11 SSTART
    [1-19:u32:404473] = 6;  // NET-IN12 +JOG
    [1-19:u32:404475] = 7;  // NET-IN13 -JOG
    [1-19:u32:404477] = 1;  // NET-IN14 FWD
    [1-19:u32:404479] = 2;  // NET-IN15 RVS

    [1-19:u32:404353] = 32; // IN0  R0
    [1-19:u32:404355] = 33; // IN1  R1
    [1-19:u32:404357] = 34; // IN2  R2
    [1-19:u32:404359] = 35; // IN3  R3
    [1-19:u32:404361] = 36; // IN4  R4
    [1-19:u32:404363] = 37; // IN5  R5
    [1-19:u32:404365] = 18; // IN6  STOP
    [1-19:u32:404367] = 39; // IN7  R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63050] = [1-19:s32:400199];
    [1-19:s32:400909] = [s32:GD63050];

    // Execution of Configuration
    [1-19:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63056] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63056] > 1 && [b:GB62204] == 1){

    [1-19:s32:400395] = 1; // Executes P-PRESET
    [1-19:s32:400909] = [s32:GD63052]; // Restore the preset position parameter after backup

```

```

[1-19:s32:400397] = 0;          // Zero clear execution of Configuration
[1-19:s32:400395] = 0;          // Zero clear for the execution of P-PRESET

[b:GB62203] = 0;                // Clear Operation preparation start trigger
                                // End the starting trigger in this script.
[b:GB62204] = 0;                // Clear Preparing operation flag
[b:GB62205] = 1;                // Touch panel in operation flag (1: in operation)
}

// 20 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62213] == 1 && [b:GB62214] == 0){

    [b:GB62214] = 1;             // Preparing operation flag (1: Preparing operation)
    bmov([1-20:u32:404449],[u32:GD63102],16); // Backs up NET-IN0-15
    bmov([1-20:u32:404353],[u32:GD63134],8);  // Backs up IN0-7
    [s32:GD63152] = [1-20:s32:400909];        // Backs up the preset position
    [u32:GD63154] = [1-20:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-20:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-20:u32:404449] = 48; // NET-IN0  M0
    [1-20:u32:404451] = 49; // NET-IN1  M1
    [1-20:u32:404453] = 50; // NET-IN2  M2
    [1-20:u32:404455] = 51; // NET-IN3  M3
    [1-20:u32:404457] = 52; // NET-IN4  M4
    [1-20:u32:404459] = 53; // NET-IN5  M5
    [1-20:u32:404461] = 4;  // NET-IN6  START
    [1-20:u32:404463] = 3;  // NET-IN7  HOME
    [1-20:u32:404465] = 18; // NET-IN8  STOP
    [1-20:u32:404467] = 9;  // NET-IN9  MS1
    [1-20:u32:404469] = 10; // NET-IN10 MS2
    [1-20:u32:404471] = 5;  // NET-IN11 SSTART
    [1-20:u32:404473] = 6;  // NET-IN12 +JOG
    [1-20:u32:404475] = 7;  // NET-IN13 -JOG
    [1-20:u32:404477] = 1;  // NET-IN14 FWD
    [1-20:u32:404479] = 2;  // NET-IN15 RVS

    [1-20:u32:404353] = 32; // IN0 R0
    [1-20:u32:404355] = 33; // IN1 R1
    [1-20:u32:404357] = 34; // IN2 R2
    [1-20:u32:404359] = 35; // IN3 R3
    [1-20:u32:404361] = 36; // IN4 R4
    [1-20:u32:404363] = 37; // IN5 R5
    [1-20:u32:404365] = 18; // IN6 STOP
    [1-20:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63150] = [1-20:s32:400199];
    [1-20:s32:400909] = [s32:GD63150];

    // Execution of Configuration
    [1-20:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63156] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63156] > 1 && [b:GB62214] == 1){

```

```

[1-20:s32:400395] = 1;           // Executes P-PRESET
[1-20:s32:400909] = [s32:GD63152]; // Restore the preset position parameter after backup
[1-20:s32:400397] = 0;           // Zero clear execution of Configuration
[1-20:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

[b:GB62213] = 0;                 // Clear Operation preparation start trigger
                                // End the starting trigger in this script.
[b:GB62214] = 0;                 // Clear Preparing operation flag
[b:GB62215] = 1;                 // Touch panel in operation flag (1: in operation)
}

```

Script No.	31034	Script Name	Script31034
Comment	17-20_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62185

```

// Script for positioning operation with touch panel operation
// Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No.
// After being changed, M0-M5 are turned ON by driver input command (400126).

// 17-20_Script in multiple operation No.31034
// Start the script in case GB62185, GB62195, GB62205, GB62215 in the 17-20_multiple operation
// preparation script (No.31033) are ON.

// [b:GB62185],[b:GB62195],[b:GB62205],[b:GB62215] : Touch panel in operation flag(start trigger of this
// script)
// Bit also works as interlock not to change the screen while touch panel in operation

// [1-17:u32:400195] - [1-20:u32:400195] : Modbus address of selected data No.
// [u32:GD62800], [u32GD62900], [GD63000], [u32:GD63100] : Device of operation No.

// Change M0-M5 of driver input command to reflect the selected operation data No.
// To ease the touch panel processing load,
// In case the currently selected data No. (400915) of the monitor and the selected operation data No. are not
// the same,
// change the driver input command (400126) to reflect a change

// 17 axis
if([b:GB62185] == 1){
    if([1-17:u32:400195] != [u32:GD62800]){
        [1-17:b:400126.b0] = [b:GD62800.b0]; // M0
        [1-17:b:400126.b1] = [b:GD62800.b1]; // M1
        [1-17:b:400126.b2] = [b:GD62800.b2]; // M2
        [1-17:b:400126.b3] = [b:GD62800.b3]; // M3
        [1-17:b:400126.b4] = [b:GD62800.b4]; // M4
        [1-17:b:400126.b5] = [b:GD62800.b5]; // M5
    }
}

// 18 axis
if([b:GB62195] == 1){
    if([1-18:u32:400195] != [u32:GD62900]){
        [1-18:b:400126.b0] = [b:GD62900.b0]; // M0
        [1-18:b:400126.b1] = [b:GD62900.b1]; // M1
        [1-18:b:400126.b2] = [b:GD62900.b2]; // M2
        [1-18:b:400126.b3] = [b:GD62900.b3]; // M3
        [1-18:b:400126.b4] = [b:GD62900.b4]; // M4
        [1-18:b:400126.b5] = [b:GD62900.b5]; // M5
    }
}

// 19 axis
if([b:GB62205] == 1){
    if([1-19:u32:400195] != [u32:GD63000]){

```

```

    [1-19:b:400126.b0] = [b:GD63000.b0]; // M0
    [1-19:b:400126.b1] = [b:GD63000.b1]; // M1
    [1-19:b:400126.b2] = [b:GD63000.b2]; // M2
    [1-19:b:400126.b3] = [b:GD63000.b3]; // M3
    [1-19:b:400126.b4] = [b:GD63000.b4]; // M4
    [1-19:b:400126.b5] = [b:GD63000.b5]; // M5
  }
}

```

// 20 axis

```
if([b:GB62215] == 1){
```

```

  if([1-20:u32:400195] != [u32:GD63100]){
    [1-20:b:400126.b0] = [b:GD63100.b0]; // M0
    [1-20:b:400126.b1] = [b:GD63100.b1]; // M1
    [1-20:b:400126.b2] = [b:GD63100.b2]; // M2
    [1-20:b:400126.b3] = [b:GD63100.b3]; // M3
    [1-20:b:400126.b4] = [b:GD63100.b4]; // M4
    [1-20:b:400126.b5] = [b:GD63100.b5]; // M5
  }
}

```

Script No.	31035	Script Name	Script31035
Comment	17-20_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62186

// Script to disable touch panel operation

// Restore such as the NET-IN signal and the necessary parameter from backup

// Screen script for 17-20 axis multiple operation

// 17-20\_End multiple operation script No.31035

// Start the script in case GB62186, GB62196, GB62206, GB62216 in the 17-20\_Check before start/end multiple operation script (No.31032) are ON.

// [b:GB62185], [b:GB62195], [b:GB62205], [b:GB62215] : Touch panel in operation flag

// [b:GB62186], [b:GB62196], [b:GB62206], [b:GB62216] : Operation-end start trigger (start trigger of this script)

// [b:GB62187], [b:GB62197], [b:GB62207], [b:GB62217] : the operation-end in process flag

// [u32:GD62802], [u32:GD62902], [u32:GD63002], [u32:GD63102] : Backup device for NET-IN0-15

// [u32:GD62834], [u32:GD62934], [u32:GD63034], [u32:GD63134] : Backup device for IN0-7

// [s32:GD62850], [s32:GD62950], [s32:GD63050], [s32:GD63150] : Backup device for Command position of a monitor

// [s32:GD62852], [s32:GD62952], [s32:GD63052], [s32:GD63152] : Backup device for preset position

// [u32:GD62854], [u32:GD62954], [u32:GD63054], [u32:GD63154] : Backup device for JOG moving distance (minimum moving distance for GOT)

// [w:GD62856], [w:GD62956], [w:GD63056], [w:GD63156] : Configuration execution waiting timer

// 17 axis

// In case Operation-end start trigger and Operation-end are not completed.

```
if([b:GB62186] == 1 && [b:GB62187] == 0){
```

```

  [b:GB62185] = 0; // Clear touch panel in operation flag
  [b:GB62187] = 1; // the operation-end in process flag (1: operation end in process)
  bmov([u32:GD62802],[1-17:u32:404449],16); // Restore NET-IN0-15 from backup
  bmov([u32:GD62834],[1-17:u32:404353],8); // Restore IN0-7 from backup
  [1-17:u32:404169] = [u32:GD62854]; // Restore JOG moving distance from backup

```

// Set the preset position parameter to the command position of the monitor

```
[s32:GD62850] = [1-17:s32:400199];
```

```
[1-17:s32:400909] = [s32:GD62850];
```

// Clear the input value of NET-IN

```
[1-17:w:400126] = 0;
```

// Execution of Configuration

```
[1-17:s32:400397] = 1;
```

```

// Start recording of Configuration execution waiting timer [sec]
[w:GD62856] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62856] > 1 && [b:GB62187] == 1){

    [1-17:s32:400395] = 1;           // Executes P-PRESET
    [1-17:s32:400909] = [s32:GD62852]; // Reset the Preset position parameter
    [1-17:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-17:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62186] = 0;                 // Clear Operation-end start trigger
                                     // End the starting trigger in this script.
    [b:GB62187] = 0;                 // Clear the operation-end in process flag
}

// 18 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62196] == 1 && [b:GB62197] == 0){

    [b:GB62195] = 0;                 // Clear touch panel in operation flag
    [b:GB62197] = 1;                 // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD62902],[1-18:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD62934],[1-18:u32:404353],8);  // Restore IN0-7 from backup
    [1-18:u32:404169] = [u32:GD62954];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD62950] = [1-18:s32:400199];
    [1-18:s32:400909] = [s32:GD62950];

    // Clear the input value of NET-IN
    [1-18:w:400126] = 0;

    // Execution of Configuration
    [1-18:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD62956] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD62956] > 1 && [b:GB62197] == 1){

    [1-18:s32:400395] = 1;           // Executes P-PRESET
    [1-18:s32:400909] = [s32:GD62952]; // Reset the Preset position parameter
    [1-18:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-18:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62196] = 0;                 // Clear Operation-end start trigger
                                     // End the starting trigger in this script.
    [b:GB62197] = 0;                 // Clear the operation-end in process flag
}

// 19 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62206] == 1 && [b:GB62207] == 0){

    [b:GB62205] = 0;                 // Clear touch panel in operation flag
    [b:GB62207] = 1;                 // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63002],[1-19:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD63034],[1-19:u32:404353],8);  // Restore IN0-7 from backup
    [1-19:u32:404169] = [u32:GD63054];        // Restore JOG moving distance from backup
}

```



```

// Set the preset position parameter to the command position of the monitor
[s32:GD63050] = [1-19:s32:400199];
[1-19:s32:400909] = [s32:GD63050];

// Clear the input value of NET-IN
[1-19:w:400126] = 0;

// Execution of Configuration
[1-19:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63056] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63056] > 1 && [b:GB62207] == 1){

    [1-19:s32:400395] = 1;           // Executes P-PRESET
    [1-19:s32:400909] = [s32:GD63052]; // Reset the Preset position parameter
    [1-19:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-19:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62206] = 0;                // Clear Operation-end start trigger
                                     // End the starting trigger in this script.
    [b:GB62207] = 0;                // Clear the operation-end in process flag
}

// 20 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62216] == 1 && [b:GB62217] == 0){

    [b:GB62215] = 0;                // Clear touch panel in operation flag
    [b:GB62217] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63102],[1-20:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD63134],[1-20:u32:404353],8);  // Restore IN0-7 from backup
    [1-20:u32:404169] = [u32:GD63154];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63150] = [1-20:s32:400199];
    [1-20:s32:400909] = [s32:GD63150];

    // Clear the input value of NET-IN
    [1-20:w:400126] = 0;

    // Execution of Configuration
    [1-20:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63156] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63156] > 1 && [b:GB62217] == 1){

    [1-20:s32:400395] = 1;           // Executes P-PRESET
    [1-20:s32:400909] = [s32:GD63152]; // Reset the Preset position parameter
    [1-20:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-20:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62216] = 0;                // Clear Operation-end start trigger
                                     // End the starting trigger in this script.
    [b:GB62217] = 0;                // Clear the operation-end in process flag
}

```

Script No.	31032	Script Name	Script31032
Comment	17-20ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62190
Same as the script No. 31032 for the base screen 31026			
Script No.	31033	Script Name	Script31033
Comment	17-20_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62193
Same as the script No. 31033 for the base screen 31026			
Script No.	31034	Script Name	Script31034
Comment	17-20_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62195
Same as the script No. 31034 for the base screen 31026			
Script No.	31035	Script Name	Script31035
Comment	17-20_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62196
Same as the script No. 31035 for the base screen 31026			
Script No.	31032	Script Name	Script31032
Comment	17-20ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62200
Same as the script No. 31032 for the base screen 31026			
Script No.	31033	Script Name	Script31033
Comment	17-20_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62203
Same as the script No. 31033 for the base screen 31026			
Script No.	31034	Script Name	Script31034
Comment	17-20_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62205
Same as the script No. 31034 for the base screen 31026			
Script No.	31035	Script Name	Script31035
Comment	17-20_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62206
Same as the script No. 31035 for the base screen 31026			
Script No.	31032	Script Name	Script31032
Comment	17-20ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62210
Same as the script No. 31032 for the base screen 31026			
Script No.	31033	Script Name	Script31033
Comment	17-20_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62213
Same as the script No. 31033 for the base screen 31026			
Script No.	31034	Script Name	Script31034
Comment	17-20_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62215

Same as the script No. 31034 for the base screen 31026

Script No.	31035	Script Name	Script31035
Comment	17-20_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62216

Same as the script No. 31035 for the base screen 31026

#### Base screen 31027

Script No.	31036	Script Name	Script31036
Comment	21-24ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62220

// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end

// Screen script for 21-24 axis multiple operation

// 21-24\_Check before start/end multiple operation script No.31036

// Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of 21-24 axis

// [b:GB62220] : 21 axis starting trigger using bit set

// [b:GB62230] : 22 axis starting trigger using bit set

// [b:GB62240] : 23 axis starting trigger using bit set

// [b:GB62250] : 24 axis starting trigger using bit set

// [b:GB62225] : 21 axis touch panel in operation flag

// [b:GB62235] : 22 axis touch panel in operation flag

// [b:GB62245] : 23 axis touch panel in operation flag

// [b:GB62255] : 24 axis touch panel in operation flag

// [w:GD61000] = 1 means as below

// Because Occurring alarm checking screen is responsible for station No. switching

// Station No.(CPU No. of driver) is needed for alarm clear (Therefore settings of each axis is required.)

// Screen switching device (GD61199) is responsible for Station No. reset

// 21 axis

if([b:GB62220] == 1){

    [w:GD61000] = 21;

    // Alarm check

    if([1-21:s32:400129] == 0){     // In case of no alarm

        [b:GB62221] = 1;     // Checking flag to see if an alarm is occurring(1: no alarm)

        [w:GD60004] = 0;     // Clear the overlap window for occurring alarm checking.

    } else {     // in case the alarm occurring

        [b:GB62221] = 0;     // Checking flag to see if an alarm is occurring(0: alarm occurring)

        [w:GD60004] = 32007;     // Display the overlap window for occurring alarm checking.

    }

    // In operation check

    if([1-21:s32:400201] == 0){     // Case: not in operation

        [b:GB62222] = 1;     // In operation checking flag (1: not in operation)

    } else {     // In case the motor is running

        [b:GB62222] = 0;     // In operation checking flag (0: in operation)

        [w:GD60004] = 32008;     // Display the overlap window to check whether a motor is in operation.

    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above

    if([b:GB62221] == 1 && [b:GB62222] == 1){

        if([b:GB62225] == 0){     // Prepare Operation start in case the touch panel is not in operation.

            [b:GB62220] = 0;     // End the starting trigger of this script

            [b:GB62223] = 1;     // Operation preparation start trigger

            [b:GB62221] = 0;     // Clear checking flag to see if an alarm is occurring

            [b:GB62222] = 0;     // Clear the In operation checking flag

```

    } else {
        [b:GB62220] = 0;    // End the starting trigger of this script
        [b:GB62226] = 1;    // Start trigger to terminate operation
        [b:GB62221] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62222] = 0;    // Clear the In operation checking flag
    }
} else {
    [b:GB62220] = 0;    // End the starting trigger of this script
    [b:GB62221] = 0;    // Clear checking flag to see if an alarm is occurring
    [b:GB62222] = 0;    // Clear the In operation checking flag
}
}

// 22 axis
if([b:GB62230] == 1){
    [w:GD61000] = 22;

    // Alarm check
    if([1-22:s32:400129] == 0){    // In case of no alarm
        [b:GB62231] = 1;    // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;    // Clear the overlap window for occurring alarm checking.
    } else {    // in case the alarm occurring
        [b:GB62231] = 0;    // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-22:s32:400201] == 0){    // Case: not in operation
        [b:GB62232] = 1;    // In operation checking flag (1: not in operation)
    } else {    // In case the motor is running
        [b:GB62232] = 0;    // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62231] == 1 && [b:GB62232] == 1){
        if([b:GB62235] == 0){    // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62230] = 0;    // End the starting trigger of this script
            [b:GB62233] = 1;    // Operation preparation start trigger
            [b:GB62231] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62232] = 0;    // Clear the In operation checking flag
        } else {    // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62230] = 0;    // End the starting trigger of this script
            [b:GB62236] = 1;    // Start trigger to terminate operation
            [b:GB62231] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62232] = 0;    // Clear the In operation checking flag
        }
    } else {
        [b:GB62230] = 0;    // End the starting trigger of this script
        [b:GB62231] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62232] = 0;    // Clear the In operation checking flag
    }
}

// 23 axis
if([b:GB62240] == 1){
    [w:GD61000] = 23;

    // Alarm check
    if([1-23:s32:400129] == 0){    // In case of no alarm
        [b:GB62241] = 1;    // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;    // Clear the overlap window for occurring alarm checking.
    } else {    // in case the alarm occurring
        [b:GB62241] = 0;    // Checking flag to see if an alarm is occurring(0: alarm occurring)
    }
}

```

```

    [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
}

// In operation check
if([1-23:s32:400201] == 0){    // Case: not in operation
    [b:GB62242] = 1;    // In operation checking flag (1: not in operation)
} else {    // In case the motor is running
    [b:GB62242] = 0;    // In operation checking flag (0: in operation)
    [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62241] == 1 && [b:GB62242] == 1){
    if([b:GB62245] == 0){    // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62240] = 0;    // End the starting trigger of this script
        [b:GB62243] = 1;    // Operation preparation start trigger
        [b:GB62241] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62242] = 0;    // Clear the In operation checking flag
    } else {    // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62240] = 0;    // End the starting trigger of this script
        [b:GB62246] = 1;    // Start trigger to terminate operation
        [b:GB62241] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62242] = 0;    // Clear the In operation checking flag
    }
} else {
    [b:GB62240] = 0;    // End the starting trigger of this script
    [b:GB62241] = 0;    // Clear checking flag to see if an alarm is occurring
    [b:GB62242] = 0;    // Clear the In operation checking flag
}
}

// 24 axis
if([b:GB62250] == 1){
    [w:GD61000] = 24;

    // Alarm check
    if([1-24:s32:400129] == 0){    // In case of no alarm
        [b:GB62251] = 1;    // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;    // Clear the overlap window for occurring alarm checking.
    } else {    // in case the alarm occurring
        [b:GB62251] = 0;    // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-24:s32:400201] == 0){    // Case: not in operation
        [b:GB62252] = 1;    // In operation checking flag (1: not in operation)
    } else {    // In case the motor is running
        [b:GB62252] = 0;    // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62251] == 1 && [b:GB62252] == 1){
        if([b:GB62255] == 0){    // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62250] = 0;    // End the starting trigger of this script
            [b:GB62253] = 1;    // Operation preparation start trigger
            [b:GB62251] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62252] = 0;    // Clear the In operation checking flag
        } else {    // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62250] = 0;    // End the starting trigger of this script
            [b:GB62256] = 1;    // Start trigger to terminate operation
            [b:GB62251] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62252] = 0;    // Clear the In operation checking flag
        }
    }
}

```

```

    }
  } else {
    [b:GB62250] = 0;    // End the starting trigger of this script
    [b:GB62251] = 0;    // Clear checking flag to see if an alarm is occurring
    [b:GB62252] = 0;    // Clear the In operation checking flag
  }
}

```

Script No.	31037	Script Name	Script31037
Comment	21-24_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62223

```

// Script to prepare for touch panel operation
// Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter
// Screen script for 21-24 axis multiple operation

// 21-24_Multiple operation preparation script No.31037
// Start the script in case GB62223, GB62233, GB62243, GB62253 in the 21-24_Check before start/end
multiple operation script (No.31036) are ON.

// [b:GB62223], [b:GB62233], [b:GB62243], [b:GB62253] : Operation preparation start trigger (start trigger of
this script)
// [b:GB62224], [b:GB62234], [b:GB62244], [b:GB62254] : Preparing operation flag
// [b:GB62225], [b:GB62235], [b:GB62245], [b:GB62255] : Touch panel in operation flag
// [u32:GD63202], [u32:GD63302], [u32:GD63402], [u32:GD63502] : Backup device for NET-IN0-15
// [u32:GD63234], [u32:GD63334], [u32:GD63434], [u32:GD63534] : Backup device for IN0-7
// [s32:GD63250], [s32:GD63350], [s32:GD63450], [s32:GD63550] : Backup device for Command position of
a monitor
// [s32:GD63252], [s32:GD63352], [s32:GD63452], [u32:GD63552] : Backup device for preset position
// [u32:GD63254],[u32:GD63354],[u32:GD63454],[u32:GD63554] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD63256], [w:GD63356], [w:GD63456], [w:GD63556] : Configuration execution waiting timer

// 21 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62223] == 1 && [b:GB62224] == 0){

  [b:GB62224] = 1;          // Preparing operation flag (1: Preparing operation)
  bmov([1-21:u32:404449],[u32:GD63202],16);    // Backs up NET-IN0-15
  bmov([1-21:u32:404353],[u32:GD63234],8);      // Backs up IN0-7
  [s32:GD63252] = [1-21:s32:400909];            // Backs up the preset position
  [u32:GD63254] = [1-21:u32:404169];            // Backup JOG moving distance (used for Minimum JOG
moving distance)

  // Clear the input value of NET-IN
  [1-21:w:400126] = 0;

  // Change the signal assignment for touch panel operation
  [1-21:u32:404449] = 48;    // NET-IN0  M0
  [1-21:u32:404451] = 49;    // NET-IN1  M1
  [1-21:u32:404453] = 50;    // NET-IN2  M2
  [1-21:u32:404455] = 51;    // NET-IN3  M3
  [1-21:u32:404457] = 52;    // NET-IN4  M4
  [1-21:u32:404459] = 53;    // NET-IN5  M5
  [1-21:u32:404461] = 4;     // NET-IN6  START
  [1-21:u32:404463] = 3;     // NET-IN7  HOME
  [1-21:u32:404465] = 18;    // NET-IN8  STOP
  [1-21:u32:404467] = 9;     // NET-IN9  MS1
  [1-21:u32:404469] = 10;    // NET-IN10 MS2
  [1-21:u32:404471] = 5;     // NET-IN11 SSTART
  [1-21:u32:404473] = 6;     // NET-IN12 +JOG
  [1-21:u32:404475] = 7;     // NET-IN13 -JOG
  [1-21:u32:404477] = 1;     // NET-IN14 FWD
  [1-21:u32:404479] = 2;     // NET-IN15 RVS

  [1-21:u32:404353] = 32;    // IN0 R0

```

```

[1-21:u32:404355] = 33; // IN1 R1
[1-21:u32:404357] = 34; // IN2 R2
[1-21:u32:404359] = 35; // IN3 R3
[1-21:u32:404361] = 36; // IN4 R4
[1-21:u32:404363] = 37; // IN5 R5
[1-21:u32:404365] = 18; // IN6 STOP
[1-21:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD63250] = [1-21:s32:400199];
[1-21:s32:400909] = [s32:GD63250];

// Execution of Configuration
[1-21:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63256] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63256] > 1 && [b:GB62224] == 1){

    [1-21:s32:400395] = 1; // Executes P-PRESET
    [1-21:s32:400909] = [s32:GD63252]; // Restore the preset position parameter after backup
    [1-21:s32:400397] = 0; // Zero clear execution of Configuration
    [1-21:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62223] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62224] = 0; // Clear Preparing operation flag
    [b:GB62225] = 1; // Touch panel in operation flag (1: in operation)
}

// 22 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62233] == 1 && [b:GB62234] == 0){

    [b:GB62234] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-22:u32:404449],[u32:GD63302],16); // Backs up NET-IN0-15
    bmov([1-22:u32:404353],[u32:GD63334],8); // Backs up IN0-7
    [s32:GD63352] = [1-22:s32:400909]; // Backs up the preset position
    [u32:GD63354] = [1-22:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-22:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-22:u32:404449] = 48; // NET-IN0 M0
    [1-22:u32:404451] = 49; // NET-IN1 M1
    [1-22:u32:404453] = 50; // NET-IN2 M2
    [1-22:u32:404455] = 51; // NET-IN3 M3
    [1-22:u32:404457] = 52; // NET-IN4 M4
    [1-22:u32:404459] = 53; // NET-IN5 M5
    [1-22:u32:404461] = 4; // NET-IN6 START
    [1-22:u32:404463] = 3; // NET-IN7 HOME
    [1-22:u32:404465] = 18; // NET-IN8 STOP
    [1-22:u32:404467] = 9; // NET-IN9 MS1
    [1-22:u32:404469] = 10; // NET-IN10 MS2
    [1-22:u32:404471] = 5; // NET-IN11 SSTART
    [1-22:u32:404473] = 6; // NET-IN12 +JOG
    [1-22:u32:404475] = 7; // NET-IN13 -JOG
    [1-22:u32:404477] = 1; // NET-IN14 FWD
    [1-22:u32:404479] = 2; // NET-IN15 RVS

```

```

[1-22:u32:404353] = 32; // IN0 R0
[1-22:u32:404355] = 33; // IN1 R1
[1-22:u32:404357] = 34; // IN2 R2
[1-22:u32:404359] = 35; // IN3 R3
[1-22:u32:404361] = 36; // IN4 R4
[1-22:u32:404363] = 37; // IN5 R5
[1-22:u32:404365] = 18; // IN6 STOP
[1-22:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD63350] = [1-22:s32:400199];
[1-22:s32:400909] = [s32:GD63350];

// Execution of Configuration
[1-22:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63356] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63356] > 1 && [b:GB62234] == 1){

    [1-22:s32:400395] = 1; // Executes P-PRESET
    [1-22:s32:400909] = [s32:GD63352]; // Restore the preset position parameter after backup
    [1-22:s32:400397] = 0; // Zero clear execution of Configuration
    [1-22:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62233] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62234] = 0; // Clear Preparing operation flag
    [b:GB62235] = 1; // Touch panel in operation flag (1: in operation)
}

// 23 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62243] == 1 && [b:GB62244] == 0){

    [b:GB62244] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-23:u32:404449],[u32:GD63402],16); // Backs up NET-IN0-15
    bmov([1-23:u32:404353],[u32:GD63434],8); // Backs up IN0-7
    [s32:GD63452] = [1-23:s32:400909]; // Backs up the preset position
    [u32:GD63454] = [1-23:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-23:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-23:u32:404449] = 48; // NET-IN0 M0
    [1-23:u32:404451] = 49; // NET-IN1 M1
    [1-23:u32:404453] = 50; // NET-IN2 M2
    [1-23:u32:404455] = 51; // NET-IN3 M3
    [1-23:u32:404457] = 52; // NET-IN4 M4
    [1-23:u32:404459] = 53; // NET-IN5 M5
    [1-23:u32:404461] = 4; // NET-IN6 START
    [1-23:u32:404463] = 3; // NET-IN7 HOME
    [1-23:u32:404465] = 18; // NET-IN8 STOP
    [1-23:u32:404467] = 9; // NET-IN9 MS1
    [1-23:u32:404469] = 10; // NET-IN10 MS2
    [1-23:u32:404471] = 5; // NET-IN11 SSTART
    [1-23:u32:404473] = 6; // NET-IN12 +JOG
    [1-23:u32:404475] = 7; // NET-IN13 -JOG

```



```

[1-23:u32:404477] = 1;    // NET-IN14 FWD
[1-23:u32:404479] = 2;    // NET-IN15 RVS

[1-23:u32:404353] = 32; // IN0 R0
[1-23:u32:404355] = 33; // IN1 R1
[1-23:u32:404357] = 34; // IN2 R2
[1-23:u32:404359] = 35; // IN3 R3
[1-23:u32:404361] = 36; // IN4 R4
[1-23:u32:404363] = 37; // IN5 R5
[1-23:u32:404365] = 18; // IN6 STOP
[1-23:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD63450] = [1-23:s32:400199];
[1-23:s32:400909] = [s32:GD63450];

// Execution of Configuration
[1-23:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63456] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63456] > 1 && [b:GB62244] == 1){

    [1-23:s32:400395] = 1;          // Executes P-PRESET
    [1-23:s32:400909] = [s32:GD63452]; // Restore the preset position parameter after backup
    [1-23:s32:400397] = 0;          // Zero clear execution of Configuration
    [1-23:s32:400395] = 0;          // Zero clear for the execution of P-PRESET

    [b:GB62243] = 0;                // Clear Operation preparation start trigger
                                   // End the starting trigger in this script.
    [b:GB62244] = 0;                // Clear Preparing operation flag
    [b:GB62245] = 1;                // Touch panel in operation flag (1: in operation)
}

// 24 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62253] == 1 && [b:GB62254] == 0){

    [b:GB62254] = 1;                // Preparing operation flag (1: Preparing operation)
    bmov([1-24:u32:404449],[u32:GD63502],16); // Backs up NET-IN0-15
    bmov([1-24:u32:404353],[u32:GD63534],8);  // Backs up IN0-7
    [s32:GD63552] = [1-24:s32:400909];        // Backs up the preset position
    [u32:GD63554] = [1-24:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-24:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-24:u32:404449] = 48; // NET-IN0  M0
    [1-24:u32:404451] = 49; // NET-IN1  M1
    [1-24:u32:404453] = 50; // NET-IN2  M2
    [1-24:u32:404455] = 51; // NET-IN3  M3
    [1-24:u32:404457] = 52; // NET-IN4  M4
    [1-24:u32:404459] = 53; // NET-IN5  M5
    [1-24:u32:404461] = 4;  // NET-IN6  START
    [1-24:u32:404463] = 3;  // NET-IN7  HOME
    [1-24:u32:404465] = 18; // NET-IN8  STOP
    [1-24:u32:404467] = 9;  // NET-IN9  MS1
    [1-24:u32:404469] = 10; // NET-IN10 MS2
    [1-24:u32:404471] = 5;  // NET-IN11 SSTART

```

```

[1-24:u32:404473] = 6;    // NET-IN12 +JOG
[1-24:u32:404475] = 7;    // NET-IN13 -JOG
[1-24:u32:404477] = 1;    // NET-IN14 FWD
[1-24:u32:404479] = 2;    // NET-IN15 RVS

[1-24:u32:404353] = 32; // IN0 R0
[1-24:u32:404355] = 33; // IN1 R1
[1-24:u32:404357] = 34; // IN2 R2
[1-24:u32:404359] = 35; // IN3 R3
[1-24:u32:404361] = 36; // IN4 R4
[1-24:u32:404363] = 37; // IN5 R5
[1-24:u32:404365] = 18; // IN6 STOP
[1-24:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD63550] = [1-24:s32:400199];
[1-24:s32:400909] = [s32:GD63550];

// Execution of Configuration
[1-24:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63556] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63556] > 1 && [b:GB62254] == 1){

    [1-24:s32:400395] = 1;           // Executes P-PRESET
    [1-24:s32:400909] = [s32:GD63552]; // Restore the preset position parameter after backup
    [1-24:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-24:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62253] = 0;                 // Clear Operation preparation start trigger
                                     // End the starting trigger in this script.
    [b:GB62254] = 0;                 // Clear Preparing operation flag
    [b:GB62255] = 1;                 // Touch panel in operation flag (1: in operation)
}

```

Script No.	31038	Script Name	Script31038
Comment	21-24_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62225

```

// Script for positioning operation with touch panel operation
// Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No.
// After being changed, M0-M5 are turned ON by driver input command (400126).

// 21-24_Script in multiple operation No.31038
// Start the script in case GB62225, GB62235, GB62245, GB62255 in the 21-24_multiple operation
// preparation script (No.31037) are ON.

// [b:GB62225],[b:GB62235],[b:GB62245],[b:GB62255] : Touch panel in operation flag(start trigger of this
// script)
// Bit also works as interlock not to change the screen while touch panel in operation

// [1-21:u32:400195] - [1-24:u32:400195] : Modbus address of selected data No.
// [u32:GD63200], [u32GD63300], [GD63400], [u32:GD63500] : Device for the operation data No.

// Change M0-M5 of driver input command to reflect the selected operation data No.
// To ease the touch panel processing load,
// In case the currently selected data No. (400915) of the monitor and the selected operation data No. are not
// the same,
// change the driver input command (400126) to reflect a change

// 21 axis
if([b:GB62225] == 1){

```

```

if([1-21:u32:400195] != [u32:GD63200]){
    [1-21:b:400126.b0] = [b:GD63200.b0];    // M0
    [1-21:b:400126.b1] = [b:GD63200.b1];    // M1
    [1-21:b:400126.b2] = [b:GD63200.b2];    // M2
    [1-21:b:400126.b3] = [b:GD63200.b3];    // M3
    [1-21:b:400126.b4] = [b:GD63200.b4];    // M4
    [1-21:b:400126.b5] = [b:GD63200.b5];    // M5
}
}

// 22 axis
if([b:GB62235] == 1){

    if([1-22:u32:400195] != [u32:GD63300]){
        [1-22:b:400126.b0] = [b:GD63300.b0];    // M0
        [1-22:b:400126.b1] = [b:GD63300.b1];    // M1
        [1-22:b:400126.b2] = [b:GD63300.b2];    // M2
        [1-22:b:400126.b3] = [b:GD63300.b3];    // M3
        [1-22:b:400126.b4] = [b:GD63300.b4];    // M4
        [1-22:b:400126.b5] = [b:GD63300.b5];    // M5
    }
}

// 23 axis
if([b:GB62245] == 1){

    if([1-23:u32:400195] != [u32:GD63400]){
        [1-23:b:400126.b0] = [b:GD63400.b0];    // M0
        [1-23:b:400126.b1] = [b:GD63400.b1];    // M1
        [1-23:b:400126.b2] = [b:GD63400.b2];    // M2
        [1-23:b:400126.b3] = [b:GD63400.b3];    // M3
        [1-23:b:400126.b4] = [b:GD63400.b4];    // M4
        [1-23:b:400126.b5] = [b:GD63400.b5];    // M5
    }
}

// 24 axis
if([b:GB62255] == 1){

    if([1-24:u32:400195] != [u32:GD63500]){
        [1-24:b:400126.b0] = [b:GD63500.b0];    // M0
        [1-24:b:400126.b1] = [b:GD63500.b1];    // M1
        [1-24:b:400126.b2] = [b:GD63500.b2];    // M2
        [1-24:b:400126.b3] = [b:GD63500.b3];    // M3
        [1-24:b:400126.b4] = [b:GD63500.b4];    // M4
        [1-24:b:400126.b5] = [b:GD63500.b5];    // M5
    }
}
}

```

Script No.	31039	Script Name	Script31039
Comment	21-24_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62226

```

// Script to disable touch panel operation
// Restore such as the NET-IN signal and the necessary parameter from backup
// Screen script for 21-24 axis multiple operation

// 21-24_End multiple operation script No.31039
// Start the script in case GB62226, GB62236, GB62246, GB62256 in the 21-24_Check before start/end
multiple operation script (No.31036) are ON.

// [b:GB62225], [b:GB62235], [b:GB62245], [b:GB62255] : Touch panel in operation flag
// [b:GB62226], [b:GB62236], [b:GB62246], [b:GB62256] : Operation-end start trigger (start trigger of this
script)
// [b:GB62227], [b:GB62237], [b:GB62247], [b:GB62257] : the operation-end in process flag

```

```

// [u32:GD63202], [u32:GD63302], [u32:GD63402], [u32:GD63502] : Backup device for NET-IN0-15
// [u32:GD63234], [u32:GD63334], [u32:GD63434], [u32:GD63534] : Backup device for IN0-7
// [s32:GD63250], [s32:GD63350], [s32:GD63450], [s32:GD63550] : Backup device for Command position of
a monitor
// [s32:GD63252], [s32:GD63352], [s32:GD63452], [s32:GD63552] : Backup device for preset position
// [u32:GD63254],[u32:GD63354],[u32:GD63454],[u32:GD63554] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD63256], [w:GD63356], [w:GD63456], [w:GD63556] : Configuration execution waiting timer

// 21 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62226] == 1 && [b:GB62227] == 0){

    [b:GB62225] = 0;                // Clear touch panel in operation flag
    [b:GB62227] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63202],[1-21:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD63234],[1-21:u32:404353],8);    // Restore IN0-7 from backup
    [1-21:u32:404169] = [u32:GD63254];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63250] = [1-21:s32:400199];
    [1-21:s32:400909] = [s32:GD63250];

    // Clear the input value of NET-IN
    [1-21:w:400126] = 0;

    // Execution of Configuration
    [1-21:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63256] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63256] > 1 && [b:GB62227] == 1){

    [1-21:s32:400395] = 1;                // Executes P-PRESET
    [1-21:s32:400909] = [s32:GD63252];    // Reset the Preset position parameter
    [1-21:s32:400397] = 0;                // Zero clear execution of Configuration
    [1-21:s32:400395] = 0;                // Zero clear for the execution of P-PRESET

    [b:GB62226] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62227] = 0;                // Clear the operation-end in process flag
}

// 22 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62236] == 1 && [b:GB62237] == 0){

    [b:GB62235] = 0;                // Clear touch panel in operation flag
    [b:GB62237] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63302],[1-22:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD63334],[1-22:u32:404353],8);    // Restore IN0-7 from backup
    [1-22:u32:404169] = [u32:GD63354];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63350] = [1-22:s32:400199];
    [1-22:s32:400909] = [s32:GD63350];

    // Clear the input value of NET-IN
    [1-22:w:400126] = 0;

    // Execution of Configuration

```

```

[1-22:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63356] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63356] > 1 && [b:GB62237] == 1){

    [1-22:s32:400395] = 1;           // Executes P-PRESET
    [1-22:s32:400909] = [s32:GD63352]; // Reset the Preset position parameter
    [1-22:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-22:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62236] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62237] = 0;                // Clear the operation-end in process flag
}

// 23 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62246] == 1 && [b:GB62247] == 0){

    [b:GB62245] = 0;                // Clear touch panel in operation flag
    [b:GB62247] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63402],[1-23:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD63434],[1-23:u32:404353],8);  // Restore IN0-7 from backup
    [1-23:u32:404169] = [u32:GD63454];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63450] = [1-23:s32:400199];
    [1-23:s32:400909] = [s32:GD63450];

    // Clear the input value of NET-IN
    [1-23:w:400126] = 0;

    // Execution of Configuration
    [1-23:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63456] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63456] > 1 && [b:GB62247] == 1){

    [1-23:s32:400395] = 1;           // Executes P-PRESET
    [1-23:s32:400909] = [s32:GD63452]; // Reset the Preset position parameter
    [1-23:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-23:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62246] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62247] = 0;                // Clear the operation-end in process flag
}

// 24 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62256] == 1 && [b:GB62257] == 0){

    [b:GB62255] = 0;                // Clear touch panel in operation flag
    [b:GB62257] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63502],[1-24:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD63534],[1-24:u32:404353],8);  // Restore IN0-7 from backup

```

```

[1-24:u32:404169] = [u32:GD63554];          // Restore JOG moving distance from backup

// Set the preset position parameter to the command position of the monitor
[s32:GD63550] = [1-24:s32:400199];
[1-24:s32:400909] = [s32:GD63550];

// Clear the input value of NET-IN
[1-24:w:400126] = 0;

// Execution of Configuration
[1-24:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63556] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63556] > 1 && [b:GB62257] == 1){

    [1-24:s32:400395] = 1;          // Executes P-PRESET
    [1-24:s32:400909] = [s32:GD63552]; // Reset the Preset position parameter
    [1-24:s32:400397] = 0;          // Zero clear execution of Configuration
    [1-24:s32:400395] = 0;          // Zero clear for the execution of P-PRESET

    [b:GB62256] = 0;              // Clear Operation-end start trigger
                                // End the starting trigger in this script.
    [b:GB62257] = 0;              // Clear the operation-end in process flag
}

```

Script No.	31036	Script Name	Script31036
Comment	21-24ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62230
Same as the script No. 31036 for the base screen 31027			
Script No.	31037	Script Name	Script31037
Comment	21-24_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62233
Same as the script No. 31037 for the base screen 31027			
Script No.	31038	Script Name	Script31038
Comment	21-24_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62235
Same as the script No. 31038 for the base screen 31027			
Script No.	31039	Script Name	Script31039
Comment	21-24_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62236
Same as the script No. 31039 for the base screen 31027			
Script No.	31036	Script Name	Script31036
Comment	21-24ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62240
Same as the script No. 31036 for the base screen 31027			
Script No.	31037	Script Name	Script31037
Comment	21-24_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62243
Same as the script No. 31037 for the base screen 31027			

Script No.	31038	Script Name	Script31038
Comment	21-24_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62245
Same as the script No. 31038 for the base screen 31027			
Script No.	31039	Script Name	Script31039
Comment	21-24_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62246
Same as the script No. 31039 for the base screen 31027			
Script No.	31036	Script Name	Script31036
Comment	21-24ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62250
Same as the script No. 31036 for the base screen 31027			
Script No.	31037	Script Name	Script31037
Comment	21-24_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62253
Same as the script No. 31037 for the base screen 31027			
Script No.	31038	Script Name	Script31038
Comment	21-24_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62255
Same as the script No. 31038 for the base screen 31027			
Script No.	31039	Script Name	Script31039
Comment	21-24_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62256
Same as the script No. 31039 for the base screen 31027			

#### Base screen 31028

Script No.	31040	Script Name	Script31040
Comment	25-28ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62260
<p>// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end</p> <p>// Screen script for 25-28 axis multiple operation</p> <p>// 25-28_Check before start/end multiple operation script No.31040</p> <p>// Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of 25-28 axis</p> <p>// [b:GB62260] : 25 axis starting trigger using bit set</p> <p>// [b:GB62270] : 26 axis starting trigger using bit set</p> <p>// [b:GB62280] : 27 axis starting trigger using bit set</p> <p>// [b:GB62290] : 28 axis starting trigger using bit set</p> <p>// [b:GB62265] : 25 axis touch panel in operation flag</p> <p>// [b:GB62275] : 26 axis touch panel in operation flag</p> <p>// [b:GB62285] : 27 axis touch panel in operation flag</p> <p>// [b:GB62295] : 28 axis touch panel in operation flag</p> <p>// [w:GD61000] = 1 means as below</p> <p>// Because Occurring alarm checking screen is responsible for station No. switching</p> <p>// Station No.(CPU No. of driver) is needed for alarm clear (Therefore settings of each axis is required.)</p> <p>// Screen switching device (GD61199) is responsible for Station No. reset</p> <p>// 25 axis</p>			

```

if([b:GB62260] == 1){
    [w:GD61000] = 25;

    // Alarm check
    if([1-25:s32:400129] == 0){ // In case of no alarm
        [b:GB62261] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62261] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }
}

// In operation check
if([1-25:s32:400201] == 0){ // Case: not in operation
    [b:GB62262] = 1; // In operation checking flag (1: not in operation)
} else { // In case the motor is running
    [b:GB62262] = 0; // In operation checking flag (0: in operation)
    [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62261] == 1 && [b:GB62262] == 1){
    if([b:GB62265] == 0){ // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62260] = 0; // End the starting trigger of this script
        [b:GB62263] = 1; // Operation preparation start trigger
        [b:GB62261] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62262] = 0; // Clear the In operation checking flag
    } else { // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62260] = 0; // End the starting trigger of this script
        [b:GB62266] = 1; // Start trigger to terminate operation
        [b:GB62261] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62262] = 0; // Clear the In operation checking flag
    }
} else {
    [b:GB62260] = 0; // End the starting trigger of this script
    [b:GB62261] = 0; // Clear checking flag to see if an alarm is occurring
    [b:GB62262] = 0; // Clear the In operation checking flag
}
}

// 26 axis
if([b:GB62270] == 1){
    [w:GD61000] = 26;

    // Alarm check
    if([1-26:s32:400129] == 0){ // In case of no alarm
        [b:GB62271] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62271] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }
}

// In operation check
if([1-26:s32:400201] == 0){ // Case: not in operation
    [b:GB62272] = 1; // In operation checking flag (1: not in operation)
} else { // In case the motor is running
    [b:GB62272] = 0; // In operation checking flag (0: in operation)
    [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62271] == 1 && [b:GB62272] == 1){
    if([b:GB62275] == 0){ // Prepare Operation start in case the touch panel is not in operation.

```



```

        [b:GB62270] = 0;    // End the starting trigger of this script
        [b:GB62273] = 1;    // Operation preparation start trigger
        [b:GB62271] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62272] = 0;    // Clear the In operation checking flag
    } else {                // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62270] = 0;    // End the starting trigger of this script
        [b:GB62276] = 1;    // Start trigger to terminate operation
        [b:GB62271] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62272] = 0;    // Clear the In operation checking flag
    }
} else {
    [b:GB62270] = 0;    // End the starting trigger of this script
    [b:GB62271] = 0;    // Clear checking flag to see if an alarm is occurring
    [b:GB62272] = 0;    // Clear the In operation checking flag
}
}

// 27 axis
if([b:GB62280] == 1){
    [w:GD61000] = 27;

    // Alarm check
    if([1-27:s32:400129] == 0){    // In case of no alarm
        [b:GB62281] = 1;    // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0;    // Clear the overlap window for occurring alarm checking.
    } else {                    // in case the alarm occurring
        [b:GB62281] = 0;    // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007;    // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-27:s32:400201] == 0){    // Case: not in operation
        [b:GB62282] = 1;    // In operation checking flag (1: not in operation)
    } else {                    // In case the motor is running
        [b:GB62282] = 0;    // In operation checking flag (0: in operation)
        [w:GD60004] = 32008;    // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62281] == 1 && [b:GB62282] == 1){
        if([b:GB62285] == 0){    // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62280] = 0;    // End the starting trigger of this script
            [b:GB62283] = 1;    // Operation preparation start trigger
            [b:GB62281] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62282] = 0;    // Clear the In operation checking flag
        } else {                // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62280] = 0;    // End the starting trigger of this script
            [b:GB62286] = 1;    // Start trigger to terminate operation
            [b:GB62281] = 0;    // Clear checking flag to see if an alarm is occurring
            [b:GB62282] = 0;    // Clear the In operation checking flag
        }
    } else {
        [b:GB62280] = 0;    // End the starting trigger of this script
        [b:GB62281] = 0;    // Clear checking flag to see if an alarm is occurring
        [b:GB62282] = 0;    // Clear the In operation checking flag
    }
}

// 28 axis
if([b:GB62290] == 1){
    [w:GD61000] = 28;

    // Alarm check
    if([1-28:s32:400129] == 0){    // In case of no alarm

```

```

[b:GB62291] = 1;      // Checking flag to see if an alarm is occurring(1: no alarm)
[w:GD60004] = 0;      // Clear the overlap window for occurring alarm checking.
} else {              // in case the alarm occurring
[b:GB62291] = 0;      // Checking flag to see if an alarm is occurring(0: alarm occurring)
[w:GD60004] = 32007;   // Display the overlap window for occurring alarm checking.
}

// In operation check
if([1-28:s32:400201] == 0){ // Case: not in operation
[b:GB62292] = 1;          // In operation checking flag (1: not in operation)
} else {                  // In case the motor is running
[b:GB62292] = 0;          // In operation checking flag (0: in operation)
[w:GD60004] = 32008;      // Display the overlap window to check whether a motor is in operation.
}

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62291] == 1 && [b:GB62292] == 1){
if([b:GB62295] == 0){ // Prepare Operation start in case the touch panel is not in operation.
[b:GB62290] = 0;     // End the starting trigger of this script
[b:GB62293] = 1;     // Operation preparation start trigger
[b:GB62291] = 0;     // Clear checking flag to see if an alarm is occurring
[b:GB62292] = 0;     // Clear the In operation checking flag
} else {             // Prepare Operation-end in case the touch panel is in operation.
[b:GB62290] = 0;     // End the starting trigger of this script
[b:GB62296] = 1;     // Start trigger to terminate operation
[b:GB62291] = 0;     // Clear checking flag to see if an alarm is occurring
[b:GB62292] = 0;     // Clear the In operation checking flag
}
} else {
[b:GB62290] = 0;     // End the starting trigger of this script
[b:GB62291] = 0;     // Clear checking flag to see if an alarm is occurring
[b:GB62292] = 0;     // Clear the In operation checking flag
}
}
}

```

Script No.	31041	Script Name	Script31041
Comment	25-28_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62263

```

// Script to prepare for touch panel operation
// Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter
// Screen script for 25-28 axis multiple operation

// 25-28_Multiple operation preparation script No.31041
// Start the script in case GB62263, GB62273, GB62283, GB62293 in the 25-28_Check before start/end
multiple operation script (No.31040) are ON.

// [b:GB62263], [b:GB62273], [b:GB62283], [b:GB62293] : Operation preparation start trigger (start trigger of
this script)
// [b:GB62264], [b:GB62274], [b:GB62284], [b:GB62294] : Preparing operation flag
// [b:GB62265], [b:GB62275], [b:GB62285], [b:GB62295] : Touch panel in operation flag
// [u32:GD63602], [u32:GD63702], [u32:GD63802], [u32:GD63902] : Backup device for NET-IN0-15
// [u32:GD63634], [u32:GD63734], [u32:GD63834], [u32:GD63934] : Backup device for IN0-7
// [s32:GD63650], [s32:GD63750], [s32:GD63850], [s32:GD63950] : Backup device for Command position of
a monitor
// [s32:GD63652], [s32:GD63752], [s32:GD63852], [u32:GD63952] : Backup device for preset position
// [u32:GD63654],[u32:GD63754],[u32:GD63854],[u32:GD63954] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD63656], [w:GD63756], [w:GD63856], [w:GD63956] : Configuration execution waiting timer

// 25 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62263] == 1 && [b:GB62264] == 0){

[b:GB62264] = 1;          // Preparing operation flag (1: Preparing operation)
bmov([1-25:u32:404449],[u32:GD63602],16); // Backs up NET-IN0-15

```

```

    bmov([1-25:u32:404353],[u32:GD63634],8);    // Backs up IN0-7
    [s32:GD63652] = [1-25:s32:400909];        // Backs up the preset position
    [u32:GD63654] = [1-25:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

// Clear the input value of NET-IN
[1-25:w:400126] = 0;

// Change the signal assignment for touch panel operation
[1-25:u32:404449] = 48;    // NET-IN0  M0
[1-25:u32:404451] = 49;    // NET-IN1  M1
[1-25:u32:404453] = 50;    // NET-IN2  M2
[1-25:u32:404455] = 51;    // NET-IN3  M3
[1-25:u32:404457] = 52;    // NET-IN4  M4
[1-25:u32:404459] = 53;    // NET-IN5  M5
[1-25:u32:404461] = 4;     // NET-IN6  START
[1-25:u32:404463] = 3;     // NET-IN7  HOME
[1-25:u32:404465] = 18;    // NET-IN8  STOP
[1-25:u32:404467] = 9;     // NET-IN9  MS1
[1-25:u32:404469] = 10;    // NET-IN10 MS2
[1-25:u32:404471] = 5;     // NET-IN11 SSTART
[1-25:u32:404473] = 6;     // NET-IN12 +JOG
[1-25:u32:404475] = 7;     // NET-IN13 -JOG
[1-25:u32:404477] = 1;     // NET-IN14 FWD
[1-25:u32:404479] = 2;     // NET-IN15 RVS

[1-25:u32:404353] = 32;    // IN0 R0
[1-25:u32:404355] = 33;    // IN1 R1
[1-25:u32:404357] = 34;    // IN2 R2
[1-25:u32:404359] = 35;    // IN3 R3
[1-25:u32:404361] = 36;    // IN4 R4
[1-25:u32:404363] = 37;    // IN5 R5
[1-25:u32:404365] = 18;    // IN6 STOP
[1-25:u32:404367] = 39;    // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD63650] = [1-25:s32:400199];
[1-25:s32:400909] = [s32:GD63650];

// Execution of Configuration
[1-25:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63656] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63656] > 1 && [b:GB62264] == 1){

    [1-25:s32:400395] = 1;    // Executes P-PRESET
    [1-25:s32:400909] = [s32:GD63652];    // Restore the preset position parameter after backup
    [1-25:s32:400397] = 0;    // Zero clear execution of Configuration
    [1-25:s32:400395] = 0;    // Zero clear for the execution of P-PRESET

    [b:GB62263] = 0;    // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62264] = 0;    // Clear Preparing operation flag
    [b:GB62265] = 1;    // Touch panel in operation flag (1: in operation)
}

// 26 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62273] == 1 && [b:GB62274] == 0){

```

```

[b:GB62274] = 1; // Preparing operation flag (1: Preparing operation)
bmov([1-26:u32:404449],[u32:GD63702],16); // Backs up NET-IN0-15
bmov([1-26:u32:404353],[u32:GD63734],8); // Backs up IN0-7
[s32:GD63752] = [1-26:s32:400909]; // Backs up the preset position
[u32:GD63754] = [1-26:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

// Clear the input value of NET-IN
[1-26:w:400126] = 0;

// Change the signal assignment for touch panel operation
[1-26:u32:404449] = 48; // NET-IN0 M0
[1-26:u32:404451] = 49; // NET-IN1 M1
[1-26:u32:404453] = 50; // NET-IN2 M2
[1-26:u32:404455] = 51; // NET-IN3 M3
[1-26:u32:404457] = 52; // NET-IN4 M4
[1-26:u32:404459] = 53; // NET-IN5 M5
[1-26:u32:404461] = 4; // NET-IN6 START
[1-26:u32:404463] = 3; // NET-IN7 HOME
[1-26:u32:404465] = 18; // NET-IN8 STOP
[1-26:u32:404467] = 9; // NET-IN9 MS1
[1-26:u32:404469] = 10; // NET-IN10 MS2
[1-26:u32:404471] = 5; // NET-IN11 SSTART
[1-26:u32:404473] = 6; // NET-IN12 +JOG
[1-26:u32:404475] = 7; // NET-IN13 -JOG
[1-26:u32:404477] = 1; // NET-IN14 FWD
[1-26:u32:404479] = 2; // NET-IN15 RVS

[1-26:u32:404353] = 32; // IN0 R0
[1-26:u32:404355] = 33; // IN1 R1
[1-26:u32:404357] = 34; // IN2 R2
[1-26:u32:404359] = 35; // IN3 R3
[1-26:u32:404361] = 36; // IN4 R4
[1-26:u32:404363] = 37; // IN5 R5
[1-26:u32:404365] = 18; // IN6 STOP
[1-26:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD63750] = [1-26:s32:400199];
[1-26:s32:400909] = [s32:GD63750];

// Execution of Configuration
[1-26:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63756] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63756] > 1 && [b:GB62274] == 1){

[1-26:s32:400395] = 1; // Executes P-PRESET
[1-26:s32:400909] = [s32:GD63752]; // Restore the preset position parameter after backup
[1-26:s32:400397] = 0; // Zero clear execution of Configuration
[1-26:s32:400395] = 0; // Zero clear for the execution of P-PRESET

[b:GB62273] = 0; // Clear Operation preparation start trigger
// End the starting trigger in this script.
[b:GB62274] = 0; // Clear Preparing operation flag
[b:GB62275] = 1; // Touch panel in operation flag (1: in operation)
}

// 27 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.

```

```

if([b:GB62283] == 1 && [b:GB62284] == 0){

    [b:GB62284] = 1;                // Preparing operation flag (1: Preparing operation)
    bmov([1-27:u32:404449],[u32:GD63802],16);    // Backs up NET-IN0-15
    bmov([1-27:u32:404353],[u32:GD63834],8);    // Backs up IN0-7
    [s32:GD63852] = [1-27:s32:400909];    // Backs up the preset position
    [u32:GD63854] = [1-27:u32:404169];    // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-27:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-27:u32:404449] = 48;    // NET-IN0  M0
    [1-27:u32:404451] = 49;    // NET-IN1  M1
    [1-27:u32:404453] = 50;    // NET-IN2  M2
    [1-27:u32:404455] = 51;    // NET-IN3  M3
    [1-27:u32:404457] = 52;    // NET-IN4  M4
    [1-27:u32:404459] = 53;    // NET-IN5  M5
    [1-27:u32:404461] = 4;    // NET-IN6  START
    [1-27:u32:404463] = 3;    // NET-IN7  HOME
    [1-27:u32:404465] = 18;    // NET-IN8  STOP
    [1-27:u32:404467] = 9;    // NET-IN9  MS1
    [1-27:u32:404469] = 10;    // NET-IN10 MS2
    [1-27:u32:404471] = 5;    // NET-IN11 SSTART
    [1-27:u32:404473] = 6;    // NET-IN12 +JOG
    [1-27:u32:404475] = 7;    // NET-IN13 -JOG
    [1-27:u32:404477] = 1;    // NET-IN14 FWD
    [1-27:u32:404479] = 2;    // NET-IN15 RVS

    [1-27:u32:404353] = 32; // IN0 R0
    [1-27:u32:404355] = 33; // IN1 R1
    [1-27:u32:404357] = 34; // IN2 R2
    [1-27:u32:404359] = 35; // IN3 R3
    [1-27:u32:404361] = 36; // IN4 R4
    [1-27:u32:404363] = 37; // IN5 R5
    [1-27:u32:404365] = 18; // IN6 STOP
    [1-27:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63850] = [1-27:s32:400199];
    [1-27:s32:400909] = [s32:GD63850];

    // Execution of Configuration
    [1-27:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63856] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63856] > 1 && [b:GB62284] == 1){

    [1-27:s32:400395] = 1;    // Executes P-PRESET
    [1-27:s32:400909] = [s32:GD63852];    // Restore the preset position parameter after backup
    [1-27:s32:400397] = 0;    // Zero clear execution of Configuration
    [1-27:s32:400395] = 0;    // Zero clear for the execution of P-PRESET

    [b:GB62283] = 0;    // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62284] = 0;    // Clear Preparing operation flag
    [b:GB62285] = 1;    // Touch panel in operation flag (1: in operation)
}

```

```

// 28 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62293] == 1 && [b:GB62294] == 0){

    [b:GB62294] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-28:u32:404449],[u32:GD63902],16); // Backs up NET-IN0-15
    bmov([1-28:u32:404353],[u32:GD63934],8); // Backs up IN0-7
    [s32:GD63952] = [1-28:s32:400909]; // Backs up the preset position
    [u32:GD63954] = [1-28:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-28:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-28:u32:404449] = 48; // NET-IN0 M0
    [1-28:u32:404451] = 49; // NET-IN1 M1
    [1-28:u32:404453] = 50; // NET-IN2 M2
    [1-28:u32:404455] = 51; // NET-IN3 M3
    [1-28:u32:404457] = 52; // NET-IN4 M4
    [1-28:u32:404459] = 53; // NET-IN5 M5
    [1-28:u32:404461] = 4; // NET-IN6 START
    [1-28:u32:404463] = 3; // NET-IN7 HOME
    [1-28:u32:404465] = 18; // NET-IN8 STOP
    [1-28:u32:404467] = 9; // NET-IN9 MS1
    [1-28:u32:404469] = 10; // NET-IN10 MS2
    [1-28:u32:404471] = 5; // NET-IN11 SSTART
    [1-28:u32:404473] = 6; // NET-IN12 +JOG
    [1-28:u32:404475] = 7; // NET-IN13 -JOG
    [1-28:u32:404477] = 1; // NET-IN14 FWD
    [1-28:u32:404479] = 2; // NET-IN15 RVS

    [1-28:u32:404353] = 32; // IN0 R0
    [1-28:u32:404355] = 33; // IN1 R1
    [1-28:u32:404357] = 34; // IN2 R2
    [1-28:u32:404359] = 35; // IN3 R3
    [1-28:u32:404361] = 36; // IN4 R4
    [1-28:u32:404363] = 37; // IN5 R5
    [1-28:u32:404365] = 18; // IN6 STOP
    [1-28:u32:404367] = 39; // IN7 R7

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63950] = [1-28:s32:400199];
    [1-28:s32:400909] = [s32:GD63950];

    // Execution of Configuration
    [1-28:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63956] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD63956] > 1 && [b:GB62294] == 1){

    [1-28:s32:400395] = 1; // Executes P-PRESET
    [1-28:s32:400909] = [s32:GD63952]; // Restore the preset position parameter after backup
    [1-28:s32:400397] = 0; // Zero clear execution of Configuration
    [1-28:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62293] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62294] = 0; // Clear Preparing operation flag
    [b:GB62295] = 1; // Touch panel in operation flag (1: in operation)
}

```

}			
Script No.	31042	Script Name	Script31042
Comment	25-28_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62265
<pre>// Script for positioning operation with touch panel operation // Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No. // After being changed, M0-M5 are turned ON by driver input command (400126).  // 25-28_Script in multiple operation No.31042 // Start the script in case GB62265, GB62275, GB62285, GB62295 in the 25-28_multiple operation preparation script (No.31041) are ON.  // [b:GB62265],[b:GB62275],[b:GB62285],[b:GB62295] : Touch panel in operation flag(start trigger of this script) // Bit also works as interlock not to change the screen while touch panel in operation  // [1-25:u32:400195] - [1-28:u32:400195] : Modbus address of selected data No. // [u32:GD63600], [u32:GD63700], [GD63800], [u32:GD63900] : Device for the operation data No.  // Change M0-M5 of driver input command to reflect the selected operation data No. // To ease the touch panel processing load, // In case the currently selected data No. (400915) of the monitor and the selected operation data No. are not the same, // change the driver input command (400126) to reflect a change  // 25 axis if([b:GB62265] == 1){     if([1-25:u32:400195] != [u32:GD63600]){         [1-25:b:400126.b0] = [b:GD63600.b0]; // M0         [1-25:b:400126.b1] = [b:GD63600.b1]; // M1         [1-25:b:400126.b2] = [b:GD63600.b2]; // M2         [1-25:b:400126.b3] = [b:GD63600.b3]; // M3         [1-25:b:400126.b4] = [b:GD63600.b4]; // M4         [1-25:b:400126.b5] = [b:GD63600.b5]; // M5     } }  // 26 axis if([b:GB62275] == 1){     if([1-26:u32:400195] != [u32:GD63700]){         [1-26:b:400126.b0] = [b:GD63700.b0]; // M0         [1-26:b:400126.b1] = [b:GD63700.b1]; // M1         [1-26:b:400126.b2] = [b:GD63700.b2]; // M2         [1-26:b:400126.b3] = [b:GD63700.b3]; // M3         [1-26:b:400126.b4] = [b:GD63700.b4]; // M4         [1-26:b:400126.b5] = [b:GD63700.b5]; // M5     } }  // 27 axis if([b:GB62285] == 1){     if([1-27:u32:400195] != [u32:GD63800]){         [1-27:b:400126.b0] = [b:GD63800.b0]; // M0         [1-27:b:400126.b1] = [b:GD63800.b1]; // M1         [1-27:b:400126.b2] = [b:GD63800.b2]; // M2         [1-27:b:400126.b3] = [b:GD63800.b3]; // M3         [1-27:b:400126.b4] = [b:GD63800.b4]; // M4         [1-27:b:400126.b5] = [b:GD63800.b5]; // M5     } }</pre>			

```

}

// 28 axis
if([b:GB62295] == 1){

    if([1-28:u32:400195] != [u32:GD63900]){
        [1-28:b:400126.b0] = [b:GD63900.b0];    // M0
        [1-28:b:400126.b1] = [b:GD63900.b1];    // M1
        [1-28:b:400126.b2] = [b:GD63900.b2];    // M2
        [1-28:b:400126.b3] = [b:GD63900.b3];    // M3
        [1-28:b:400126.b4] = [b:GD63900.b4];    // M4
        [1-28:b:400126.b5] = [b:GD63900.b5];    // M5
    }
}
}

```

Script No.	31043	Script Name	Script31043
Comment	25-28_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62266

```

// Script to disable touch panel operation
// Restore such as the NET-IN signal and the necessary parameter from backup
// Screen script for 25-28 axis multiple operation

// 25-28_End multiple operation script No.31043
// Start the script in case GB62266, GB62276, GB62286, GB62296 in the 25-28_Check before start/end
multiple operation script (No.31040) are ON.

// [b:GB62265], [b:GB62275], [b:GB62285], [b:GB62295] : Touch panel in operation flag
// [b:GB62266], [b:GB62276], [b:GB62286], [b:GB62296] : Operation-end start trigger (start trigger of this
script)
// [b:GB62267], [b:GB62277], [b:GB62287], [b:GB62297] : the operation-end in process flag
// [u32:GD63602], [u32:GD63702], [u32:GD63802], [u32:GD63902] : Backup device for NET-IN0-15
// [u32:GD63634], [u32:GD63734], [u32:GD63834], [u32:GD63934] : Backup device for IN0-7
// [s32:GD63650], [s32:GD63750], [s32:GD63850], [s32:GD63950] : Backup device for Command position of
a monitor
// [s32:GD63652], [s32:GD63752], [s32:GD63852], [s32:GD63952] : Backup device for preset position
// [u32:GD63654],[u32:GD63754],[u32:GD63854],[u32:GD63954] : Backup device for JOG moving distance
(minimum moving distance for GOT)
// [w:GD63656], [w:GD63756], [w:GD63856], [w:GD63956] : Configuration execution waiting timer

// 25 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62266] == 1 && [b:GB62267] == 0){

    [b:GB62265] = 0;                // Clear touch panel in operation flag
    [b:GB62267] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63602],[1-25:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD63634],[1-25:u32:404353],8);      // Restore IN0-7 from backup
    [1-25:u32:404169] = [u32:GD63654];            // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63650] = [1-25:s32:400199];
    [1-25:s32:400909] = [s32:GD63650];

    // Clear the input value of NET-IN
    [1-25:w:400126] = 0;

    // Execution of Configuration
    [1-25:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63656] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63656] > 1 && [b:GB62267] == 1){

```



```

[1-25:s32:400395] = 1;           // Executes P-PRESET
[1-25:s32:400909] = [s32:GD63652]; // Reset the Preset position parameter
[1-25:s32:400397] = 0;           // Zero clear execution of Configuration
[1-25:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

[b:GB62266] = 0;                // Clear Operation-end start trigger
                                // End the starting trigger in this script.
[b:GB62267] = 0;                // Clear the operation-end in process flag
}

// 26 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62276] == 1 && [b:GB62277] == 0){

    [b:GB62275] = 0;             // Clear touch panel in operation flag
    [b:GB62277] = 1;             // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63702],[1-26:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD63734],[1-26:u32:404353],8);  // Restore IN0-7 from backup
    [1-26:u32:404169] = [u32:GD63754];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63750] = [1-26:s32:400199];
    [1-26:s32:400909] = [s32:GD63750];

    // Clear the input value of NET-IN
    [1-26:w:400126] = 0;

    // Execution of Configuration
    [1-26:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63756] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63756] > 1 && [b:GB62277] == 1){

    [1-26:s32:400395] = 1;           // Executes P-PRESET
    [1-26:s32:400909] = [s32:GD63752]; // Reset the Preset position parameter
    [1-26:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-26:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62276] = 0;                // Clear Operation-end start trigger
                                // End the starting trigger in this script.
    [b:GB62277] = 0;                // Clear the operation-end in process flag
}

// 27 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62286] == 1 && [b:GB62287] == 0){

    [b:GB62285] = 0;             // Clear touch panel in operation flag
    [b:GB62287] = 1;             // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63802],[1-27:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD63834],[1-27:u32:404353],8);  // Restore IN0-7 from backup
    [1-27:u32:404169] = [u32:GD63854];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63850] = [1-27:s32:400199];
    [1-27:s32:400909] = [s32:GD63850];

    // Clear the input value of NET-IN
    [1-27:w:400126] = 0;

```

```

// Execution of Configuration
[1-27:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD63856] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63856] > 1 && [b:GB62287] == 1){

    [1-27:s32:400395] = 1;           // Executes P-PRESET
    [1-27:s32:400909] = [s32:GD63852]; // Reset the Preset position parameter
    [1-27:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-27:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62286] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62287] = 0;                // Clear the operation-end in process flag
}

// 28 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62296] == 1 && [b:GB62297] == 0){

    [b:GB62295] = 0;                // Clear touch panel in operation flag
    [b:GB62297] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD63902],[1-28:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD63934],[1-28:u32:404353],8); // Restore IN0-7 from backup
    [1-28:u32:404169] = [u32:GD63954]; // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD63950] = [1-28:s32:400199];
    [1-28:s32:400909] = [s32:GD63950];

    // Clear the input value of NET-IN
    [1-28:w:400126] = 0;

    // Execution of Configuration
    [1-28:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD63956] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD63956] > 1 && [b:GB62297] == 1){

    [1-28:s32:400395] = 1;           // Executes P-PRESET
    [1-28:s32:400909] = [s32:GD63952]; // Reset the Preset position parameter
    [1-28:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-28:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62296] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62297] = 0;                // Clear the operation-end in process flag
}

```

Script No.	31040	Script Name	Script31040
Comment	25-28ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62270
Same as the script No. 31040 for the base screen 31028			
Script No.	31041	Script Name	Script31041

Comment	25-28_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62273
Same as the script No. 31041 for the base screen 31028			
Script No.	31042	Script Name	Script31042
Comment	25-28_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62275
Same as the script No. 31042 for the base screen 31028			
Script No.	31043	Script Name	Script31043
Comment	25-28_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62276
Same as the script No. 31043 for the base screen 31028			
Script No.	31040	Script Name	Script31040
Comment	25-28ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62280
Same as the script No. 31040 for the base screen 31028			
Script No.	31041	Script Name	Script31041
Comment	25-28_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62283
Same as the script No. 31041 for the base screen 31028			
Script No.	31042	Script Name	Script31042
Comment	25-28_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62285
Same as the script No. 31042 for the base screen 31028			
Script No.	31043	Script Name	Script31043
Comment	25-28_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62286
Same as the script No. 31043 for the base screen 31028			
Script No.	31040	Script Name	Script31040
Comment	25-28ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62290
Same as the script No. 31040 for the base screen 31028			
Script No.	31041	Script Name	Script31041
Comment	25-28_Multi ope prep		
Data Type	Signed BIN16	Trigger Type	ON GB62293
Same as the script No. 31041 for the base screen 31028			
Script No.	31042	Script Name	Script31042
Comment	25-28_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62295
Same as the script No. 31042 for the base screen 31028			
Script No.	31043	Script Name	Script31043
Comment	25-28_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62296
Same as the script No. 31043 for the base screen 31028			

## Base screen 31029

Script No.	31044	Script Name	Script31044
Comment	29-31ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62300
<pre>// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end // Screen script for 29-31 axis multiple operation  // 29-31_Check before start/end multiple operation script No.31044 // Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of 29-31 axis  // [b:GB62300] : 29 axis starting trigger using bit set // [b:GB62310] : 30 axis starting trigger using bit set // [b:GB62320] : 31 axis starting trigger using bit set // [b:GB62305] : 29 axis touch panel in operation flag // [b:GB62315] : 30 axis touch panel in operation flag // [b:GB62325] : 31 axis touch panel in operation flag  // [w:GD61000] = 1 means as below // Because Occurring alarm checking screen is responsible for station No. switching // Station No.(CPU No. of driver) is needed for alarm clear (Therefore settings of each axis is required.) // Screen switching device (GD61199) is responsible for Station No. reset  // 29 axis if([b:GB62300] == 1){     [w:GD61000] = 29;      // Alarm check     if([1-29:s32:400129] == 0){ // In case of no alarm         [b:GB62301] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)         [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.     } else { // in case the alarm occurring         [b:GB62301] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)         [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.     }      // In operation check     if([1-29:s32:400201] == 0){ // Case: not in operation         [b:GB62302] = 1; // In operation checking flag (1: not in operation)     } else { // In case the motor is running         [b:GB62302] = 0; // In operation checking flag (0: in operation)         [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.     }      // Distinguish Operation start and Operation-end after finishing the check mentioned above     if([b:GB62301] == 1 &amp;&amp; [b:GB62302] == 1){         if([b:GB62305] == 0){ // Prepare Operation start in case the touch panel is not in operation.             [b:GB62300] = 0; // End the starting trigger of this script             [b:GB62303] = 1; // Operation preparation start trigger             [b:GB62301] = 0; // Clear checking flag to see if an alarm is occurring             [b:GB62302] = 0; // Clear the In operation checking flag         } else { // Prepare Operation-end in case the touch panel is in operation.             [b:GB62300] = 0; // End the starting trigger of this script             [b:GB62306] = 1; // Start trigger to terminate operation             [b:GB62301] = 0; // Clear checking flag to see if an alarm is occurring             [b:GB62302] = 0; // Clear the In operation checking flag         }     } else {         [b:GB62300] = 0; // End the starting trigger of this script         [b:GB62301] = 0; // Clear checking flag to see if an alarm is occurring         [b:GB62302] = 0; // Clear the In operation checking flag     } }</pre>			

```

}

// 30 axis
if([b:GB62310] == 1){
    [w:GD61000] = 30;

    // Alarm check
    if([1-30:s32:400129] == 0){ // In case of no alarm
        [b:GB62311] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62311] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-30:s32:400201] == 0){ // Case: not in operation
        [b:GB62312] = 1; // In operation checking flag (1: not in operation)
    } else { // In case the motor is running
        [b:GB62312] = 0; // In operation checking flag (0: in operation)
        [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.
    }

    // Distinguish Operation start and Operation-end after finishing the check mentioned above
    if([b:GB62311] == 1 && [b:GB62312] == 1){
        if([b:GB62315] == 0){ // Prepare Operation start in case the touch panel is not in operation.
            [b:GB62310] = 0; // End the starting trigger of this script
            [b:GB62313] = 1; // Operation preparation start trigger
            [b:GB62311] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62312] = 0; // Clear the In operation checking flag
        } else { // Prepare Operation-end in case the touch panel is in operation.
            [b:GB62310] = 0; // End the starting trigger of this script
            [b:GB62316] = 1; // Start trigger to terminate operation
            [b:GB62311] = 0; // Clear checking flag to see if an alarm is occurring
            [b:GB62312] = 0; // Clear the In operation checking flag
        }
    } else {
        [b:GB62310] = 0; // End the starting trigger of this script
        [b:GB62311] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62312] = 0; // Clear the In operation checking flag
    }
}

// 31 axis
if([b:GB62320] == 1){
    [w:GD61000] = 31;

    // Alarm check
    if([1-31:s32:400129] == 0){ // In case of no alarm
        [b:GB62321] = 1; // Checking flag to see if an alarm is occurring(1: no alarm)
        [w:GD60004] = 0; // Clear the overlap window for occurring alarm checking.
    } else { // in case the alarm occurring
        [b:GB62321] = 0; // Checking flag to see if an alarm is occurring(0: alarm occurring)
        [w:GD60004] = 32007; // Display the overlap window for occurring alarm checking.
    }

    // In operation check
    if([1-31:s32:400201] == 0){ // Case: not in operation
        [b:GB62322] = 1; // In operation checking flag (1: not in operation)
    } else { // In case the motor is running
        [b:GB62322] = 0; // In operation checking flag (0: in operation)
        [w:GD60004] = 32008; // Display the overlap window to check whether a motor is in operation.
    }
}

```

```

// Distinguish Operation start and Operation-end after finishing the check mentioned above
if([b:GB62321] == 1 && [b:GB62322] == 1){
    if([b:GB62325] == 0){ // Prepare Operation start in case the touch panel is not in operation.
        [b:GB62320] = 0; // End the starting trigger of this script
        [b:GB62323] = 1; // Operation preparation start trigger
        [b:GB62321] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62322] = 0; // Clear the In operation checking flag
    } else { // Prepare Operation-end in case the touch panel is in operation.
        [b:GB62320] = 0; // End the starting trigger of this script
        [b:GB62326] = 1; // Start trigger to terminate operation
        [b:GB62321] = 0; // Clear checking flag to see if an alarm is occurring
        [b:GB62322] = 0; // Clear the In operation checking flag
    }
} else {
    [b:GB62320] = 0; // End the starting trigger of this script
    [b:GB62321] = 0; // Clear checking flag to see if an alarm is occurring
    [b:GB62322] = 0; // Clear the In operation checking flag
}
}

```

Script No.	31045	Script Name	Script31045
Comment	29-31_Multi operation prep		
Data Type	Signed BIN16	Trigger Type	ON GB62303

```

// Script to prepare for touch panel operation
// Change the assignment of the necessary signal, such as NET-IN and backup the necessary parameter
// Screen script for 29-31 axis multiple operation

// 29-31_Multiple operation preparation script No.31045
// Start the script in case GB62303, GB62313, GB62323 in the 29-31_Check before start/end multiple
operation script (No.31044) are ON.

// [b:GB62303], [b:GB62313], [b:GB62323] : Operation preparation start trigger (start trigger of this script)
// [b:GB62304], [b:GB62314], [b:GB62324] : Preparing operation flag
// [b:GB62305], [b:GB62315], [b:GB62325] : Touch panel in operation flag
// [u32:GD64002], [u32:GD64102], [u32:GD64202] : Backup device for NET-IN0-15
// [u32:GD64034], [u32:GD64134], [u32:GD64234] : Backup device for IN0-7
// [s32:GD64050], [s32:GD64150], [s32:GD64250] : Backup device for Command position of a monitor
// [s32:GD64052], [s32:GD64152], [s32:GD64252] : Backup device for preset position
// [u32:GD64054],[u32:GD64154],[u32:GD64254] : Backup device for JOG moving distance (minimum
moving distance for GOT)
// [w:GD64056], [w:GD64156], [w:GD64256] : Configuration execution waiting timer

// 29 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62303] == 1 && [b:GB62304] == 0){

    [b:GB62304] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-29:u32:404449],[u32:GD64002],16); // Backs up NET-IN0-15
    bmov([1-29:u32:404353],[u32:GD64034],8); // Backs up IN0-7
    [s32:GD64052] = [1-29:s32:400909]; // Backs up the preset position
    [u32:GD64054] = [1-29:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-29:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-29:u32:404449] = 48; // NET-IN0 M0
    [1-29:u32:404451] = 49; // NET-IN1 M1
    [1-29:u32:404453] = 50; // NET-IN2 M2
    [1-29:u32:404455] = 51; // NET-IN3 M3
    [1-29:u32:404457] = 52; // NET-IN4 M4
    [1-29:u32:404459] = 53; // NET-IN5 M5
    [1-29:u32:404461] = 4; // NET-IN6 START
    [1-29:u32:404463] = 3; // NET-IN7 HOME

```

```

[1-29:u32:404465] = 18; // NET-IN8 STOP
[1-29:u32:404467] = 9; // NET-IN9 MS1
[1-29:u32:404469] = 10; // NET-IN10 MS2
[1-29:u32:404471] = 5; // NET-IN11 SSTART
[1-29:u32:404473] = 6; // NET-IN12 +JOG
[1-29:u32:404475] = 7; // NET-IN13 -JOG
[1-29:u32:404477] = 1; // NET-IN14 FWD
[1-29:u32:404479] = 2; // NET-IN15 RVS

[1-29:u32:404353] = 32; // IN0 R0
[1-29:u32:404355] = 33; // IN1 R1
[1-29:u32:404357] = 34; // IN2 R2
[1-29:u32:404359] = 35; // IN3 R3
[1-29:u32:404361] = 36; // IN4 R4
[1-29:u32:404363] = 37; // IN5 R5
[1-29:u32:404365] = 18; // IN6 STOP
[1-29:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD64050] = [1-29:s32:400199];
[1-29:s32:400909] = [s32:GD64050];

// Execution of Configuration
[1-29:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD64056] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD64056] > 1 && [b:GB62304] == 1){

    [1-29:s32:400395] = 1; // Executes P-PRESET
    [1-29:s32:400909] = [s32:GD64052]; // Restore the preset position parameter after backup
    [1-29:s32:400397] = 0; // Zero clear execution of Configuration
    [1-29:s32:400395] = 0; // Zero clear for the execution of P-PRESET

    [b:GB62303] = 0; // Clear Operation preparation start trigger
    // End the starting trigger in this script.
    [b:GB62304] = 0; // Clear Preparing operation flag
    [b:GB62305] = 1; // Touch panel in operation flag (1: in operation)
}

// 30 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62313] == 1 && [b:GB62314] == 0){

    [b:GB62314] = 1; // Preparing operation flag (1: Preparing operation)
    bmov([1-30:u32:404449],[u32:GD64102],16); // Backs up NET-IN0-15
    bmov([1-30:u32:404353],[u32:GD64134],8); // Backs up IN0-7
    [s32:GD64152] = [1-30:s32:400909]; // Backs up the preset position
    [u32:GD64154] = [1-30:u32:404169]; // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-30:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-30:u32:404449] = 48; // NET-IN0 M0
    [1-30:u32:404451] = 49; // NET-IN1 M1
    [1-30:u32:404453] = 50; // NET-IN2 M2
    [1-30:u32:404455] = 51; // NET-IN3 M3
    [1-30:u32:404457] = 52; // NET-IN4 M4
    [1-30:u32:404459] = 53; // NET-IN5 M5

```

```

[1-30:u32:404461] = 4;    // NET-IN6  START
[1-30:u32:404463] = 3;    // NET-IN7  HOME
[1-30:u32:404465] = 18;   // NET-IN8  STOP
[1-30:u32:404467] = 9;    // NET-IN9  MS1
[1-30:u32:404469] = 10;   // NET-IN10 MS2
[1-30:u32:404471] = 5;    // NET-IN11 SSTART
[1-30:u32:404473] = 6;    // NET-IN12 +JOG
[1-30:u32:404475] = 7;    // NET-IN13 -JOG
[1-30:u32:404477] = 1;    // NET-IN14 FWD
[1-30:u32:404479] = 2;    // NET-IN15 RVS

[1-30:u32:404353] = 32; // IN0 R0
[1-30:u32:404355] = 33; // IN1 R1
[1-30:u32:404357] = 34; // IN2 R2
[1-30:u32:404359] = 35; // IN3 R3
[1-30:u32:404361] = 36; // IN4 R4
[1-30:u32:404363] = 37; // IN5 R5
[1-30:u32:404365] = 18; // IN6 STOP
[1-30:u32:404367] = 39; // IN7 R7

// Set the preset position parameter to the command position of the monitor
[s32:GD64150] = [1-30:s32:400199];
[1-30:s32:400909] = [s32:GD64150];

// Execution of Configuration
[1-30:s32:400397] = 1;

// Start recording of Configuration execution waiting timer [sec]
[w:GD64156] = [w:GS7];
}

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD64156] > 1 && [b:GB62314] == 1){

    [1-30:s32:400395] = 1;    // Executes P-PRESET
    [1-30:s32:400909] = [s32:GD64152]; // Restore the preset position parameter after backup
    [1-30:s32:400397] = 0;    // Zero clear execution of Configuration
    [1-30:s32:400395] = 0;    // Zero clear for the execution of P-PRESET

    [b:GB62313] = 0;          // Clear Operation preparation start trigger
                                // End the starting trigger in this script.
    [b:GB62314] = 0;          // Clear Preparing operation flag
    [b:GB62315] = 1;          // Touch panel in operation flag (1: in operation)
}

// 31 axis
// In case Operation-preparation start trigger and Operation-preparation are not completed.
if([b:GB62323] == 1 && [b:GB62324] == 0){

    [b:GB62324] = 1;          // Preparing operation flag (1: Preparing operation)
    bmov([1-31:u32:404449],[u32:GD64202],16); // Backs up NET-IN0-15
    bmov([1-31:u32:404353],[u32:GD64234],8);  // Backs up IN0-7
    [s32:GD64252] = [1-31:s32:400909];        // Backs up the preset position
    [u32:GD64254] = [1-31:u32:404169];        // Backup JOG moving distance (used for Minimum JOG
moving distance)

    // Clear the input value of NET-IN
    [1-31:w:400126] = 0;

    // Change the signal assignment for touch panel operation
    [1-31:u32:404449] = 48; // NET-IN0  M0
    [1-31:u32:404451] = 49; // NET-IN1  M1
    [1-31:u32:404453] = 50; // NET-IN2  M2
    [1-31:u32:404455] = 51; // NET-IN3  M3

```



```

[1-31:u32:404457] = 52; // NET-IN4 M4
[1-31:u32:404459] = 53; // NET-IN5 M5
[1-31:u32:404461] = 4; // NET-IN6 START
[1-31:u32:404463] = 3; // NET-IN7 HOME
[1-31:u32:404465] = 18; // NET-IN8 STOP
[1-31:u32:404467] = 9; // NET-IN9 MS1
[1-31:u32:404469] = 10; // NET-IN10 MS2
[1-31:u32:404471] = 5; // NET-IN11 SSTART
[1-31:u32:404473] = 6; // NET-IN12 +JOG
[1-31:u32:404475] = 7; // NET-IN13 -JOG
[1-31:u32:404477] = 1; // NET-IN14 FWD
[1-31:u32:404479] = 2; // NET-IN15 RVS

```

```

[1-31:u32:404353] = 32; // IN0 R0
[1-31:u32:404355] = 33; // IN1 R1
[1-31:u32:404357] = 34; // IN2 R2
[1-31:u32:404359] = 35; // IN3 R3
[1-31:u32:404361] = 36; // IN4 R4
[1-31:u32:404363] = 37; // IN5 R5
[1-31:u32:404365] = 18; // IN6 STOP
[1-31:u32:404367] = 39; // IN7 R7

```

```

// Set the preset position parameter to the command position of the monitor
[s32:GD64250] = [1-31:s32:400199];
[1-31:s32:400909] = [s32:GD64250];

```

```

// Execution of Configuration
[1-31:s32:400397] = 1;

```

```

// Start recording of Configuration execution waiting timer [sec]
[w:GD64256] = [w:GS7];

```

```

}

```

```

// In case Configuration execution and Operation preparation are completed.
if([w:GS7] - [w:GD64256] > 1 && [b:GB62324] == 1){

```

```

[1-31:s32:400395] = 1; // Executes P-PRESET
[1-31:s32:400909] = [s32:GD64152]; // Restore the preset position parameter after backup
[1-31:s32:400397] = 0; // Zero clear execution of Configuration
[1-31:s32:400395] = 0; // Zero clear for the execution of P-PRESET

```

```

[b:GB62323] = 0; // Clear Operation preparation start trigger
// End the starting trigger in this script.
[b:GB62324] = 0; // Clear Preparing operation flag
[b:GB62325] = 1; // Touch panel in operation flag (1: in operation)
}

```

Script No.	31046	Script Name	Script31046
Comment	29-31_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62305

```

// Script for positioning operation with touch panel operation
// Change M0-M5 (reassigned NET-IN) to reflect the selected operation data No.
// After being changed, M0-M5 are turned ON by driver input command (400126).

// 29-31_Script in multiple operation No.31046
// Start the script in case GB62305, GB62315, GB62325 in the 29-31_multiple operation preparation script
(No.31045) are ON.

// [b:GB62305],[b:GB62315],[b:GB62325] : Touch panel in operation flag(start trigger of this script)
// Bit also works as interlock not to change the screen while touch panel in operation

// [1-29:u32:400195] - [1-31:u32:400195] : Modbus address of selected data No.
// [u32:GD64000], [u32GD64100], [GD64200] : Device for the operation data No.

// Change M0-M5 of driver input command to reflect the selected operation data No.

```

```

// To ease the touch panel processing load,
// In case the currently selected data No. (400915) of the monitor and the selected operation data No. are not
the same,
// change the driver input command (400126) to reflect a change

// 29 axis
if([b:GB62305] == 1){

    if([1-29:u32:400195] != [u32:GD64000]){
        [1-29:b:400126.b0] = [b:GD64000.b0]; // M0
        [1-29:b:400126.b1] = [b:GD64000.b1]; // M1
        [1-29:b:400126.b2] = [b:GD64000.b2]; // M2
        [1-29:b:400126.b3] = [b:GD64000.b3]; // M3
        [1-29:b:400126.b4] = [b:GD64000.b4]; // M4
        [1-29:b:400126.b5] = [b:GD64000.b5]; // M5
    }
}

// 30 axis
if([b:GB62315] == 1){

    if([1-30:u32:400195] != [u32:GD64100]){
        [1-30:b:400126.b0] = [b:GD64100.b0]; // M0
        [1-30:b:400126.b1] = [b:GD64100.b1]; // M1
        [1-30:b:400126.b2] = [b:GD64100.b2]; // M2
        [1-30:b:400126.b3] = [b:GD64100.b3]; // M3
        [1-30:b:400126.b4] = [b:GD64100.b4]; // M4
        [1-30:b:400126.b5] = [b:GD64100.b5]; // M5
    }
}

// 31 axis
if([b:GB62325] == 1){

    if([1-31:u32:400195] != [u32:GD64200]){
        [1-31:b:400126.b0] = [b:GD64200.b0]; // M0
        [1-31:b:400126.b1] = [b:GD64200.b1]; // M1
        [1-31:b:400126.b2] = [b:GD64200.b2]; // M2
        [1-31:b:400126.b3] = [b:GD64200.b3]; // M3
        [1-31:b:400126.b4] = [b:GD64200.b4]; // M4
        [1-31:b:400126.b5] = [b:GD64200.b5]; // M5
    }
}

```

Script No.	31047	Script Name	Script31047
Comment	29-31_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62306

```

// Script to disable touch panel operation
// Restore such as the NET-IN signal and the necessary parameter from backup
// Screen script for 29-31 axis multiple operation

// 29-31_End multiple operation script No.31047
// Start the script in case GB62306, GB62316, GB62326 in the 29-31_Check before start/end multiple
operation script (No.31044) are ON.

// [b:GB62305], [b:GB62315], [b:GB62325] : Touch panel in operation flag
// [b:GB62306], [b:GB62316], [b:GB62326] : Operation-end start trigger (start trigger of this script)
// [b:GB62307], [b:GB62317], [b:GB62327] : the operation-end in process flag
// [u32:GD64002], [u32:GD64102], [u32:GD64202] : Backup device for NET-IN0-15
// [u32:GD64034], [u32:GD64134], [u32:GD64234] : Backup device for IN0-7
// [s32:GD64050], [s32:GD64150], [s32:GD64250] : Backup device for Command position of a monitor
// [s32:GD64052], [s32:GD64152], [s32:GD64252] : Backup device for preset position
// [u32:GD64054],[u32:GD64154],[u32:GD64254] : Backup device for JOG moving distance (minimum
moving distance for GOT)
// [w:GD64056], [w:GD64156], [w:GD64256] : Configuration execution waiting timer

```

```

// 29 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62306] == 1 && [b:GB62307] == 0){

    [b:GB62305] = 0;                // Clear touch panel in operation flag
    [b:GB62307] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD64002],[1-29:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD64034],[1-29:u32:404353],8);    // Restore IN0-7 from backup
    [1-29:u32:404169] = [u32:GD64054];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD64050] = [1-29:s32:400199];
    [1-29:s32:400909] = [s32:GD64050];

    // Clear the input value of NET-IN
    [1-29:w:400126] = 0;

    // Execution of Configuration
    [1-29:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD64056] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD64056] > 1 && [b:GB62307] == 1){

    [1-29:s32:400395] = 1;                // Executes P-PRESET
    [1-29:s32:400909] = [s32:GD64052];    // Reset the Preset position parameter
    [1-29:s32:400397] = 0;                // Zero clear execution of Configuration
    [1-29:s32:400395] = 0;                // Zero clear for the execution of P-PRESET

    [b:GB62306] = 0;                // Clear Operation-end start trigger
    // End the starting trigger in this script.
    [b:GB62307] = 0;                // Clear the operation-end in process flag
}

// 30 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62316] == 1 && [b:GB62317] == 0){

    [b:GB62315] = 0;                // Clear touch panel in operation flag
    [b:GB62317] = 1;                // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD64102],[1-30:u32:404449],16);    // Restore NET-IN0-15 from backup
    bmov([u32:GD64134],[1-30:u32:404353],8);    // Restore IN0-7 from backup
    [1-30:u32:404169] = [u32:GD64154];    // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD64150] = [1-30:s32:400199];
    [1-30:s32:400909] = [s32:GD64150];

    // Clear the input value of NET-IN
    [1-30:w:400126] = 0;

    // Execution of Configuration
    [1-30:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD64156] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD64156] > 1 && [b:GB62317] == 1){

```

```

[1-30:s32:400395] = 1;           // Executes P-PRESET
[1-30:s32:400909] = [s32:GD64152]; // Reset the Preset position parameter
[1-30:s32:400397] = 0;           // Zero clear execution of Configuration
[1-30:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

[b:GB62316] = 0;                // Clear Operation-end start trigger
                                // End the starting trigger in this script.
[b:GB62317] = 0;                // Clear the operation-end in process flag
}

// 31 axis
// In case Operation-end start trigger and Operation-end are not completed.
if([b:GB62326] == 1 && [b:GB62327] == 0){

    [b:GB62325] = 0;             // Clear touch panel in operation flag
    [b:GB62327] = 1;             // the operation-end in process flag (1: operation end in process)
    bmov([u32:GD64202],[1-31:u32:404449],16); // Restore NET-IN0-15 from backup
    bmov([u32:GD64234],[1-31:u32:404353],8);  // Restore IN0-7 from backup
    [1-31:u32:404169] = [u32:GD64254];        // Restore JOG moving distance from backup

    // Set the preset position parameter to the command position of the monitor
    [s32:GD64250] = [1-31:s32:400199];
    [1-31:s32:400909] = [s32:GD64250];

    // Clear the input value of NET-IN
    [1-31:w:400126] = 0;

    // Execution of Configuration
    [1-31:s32:400397] = 1;

    // Start recording of Configuration execution waiting timer [sec]
    [w:GD64256] = [w:GS7];
}

// In case Configuration execution and Operation-end are completed.
if([w:GS7] - [w:GD64256] > 1 && [b:GB62327] == 1){

    [1-31:s32:400395] = 1;           // Executes P-PRESET
    [1-31:s32:400909] = [s32:GD64252]; // Reset the Preset position parameter
    [1-31:s32:400397] = 0;           // Zero clear execution of Configuration
    [1-31:s32:400395] = 0;           // Zero clear for the execution of P-PRESET

    [b:GB62326] = 0;                // Clear Operation-end start trigger
                                // End the starting trigger in this script.
    [b:GB62327] = 0;                // Clear the operation-end in process flag
}

```

Script No.	31044	Script Name	Script31044
Comment	29-31ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62310
Same as the script No. 31044 for the base screen 31029			
Script No.	31045	Script Name	Script31045
Comment	29-31_Multi operation prep		
Data Type	Signed BIN16	Trigger Type	ON GB62313
Same as the script No. 31045 for the base screen 31029			
Script No.	31046	Script Name	Script31046
Comment	29-31_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62315

Same as the script No. 31046 for the base screen 31029			
Script No.	31047	Script Name	Script31047
Comment	29-31_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62316
Same as the script No. 31047 for the base screen 31029			
Script No.	31044	Script Name	Script31044
Comment	29-31ChkBeforeStartEndOfMultiOpe		
Data Type	Signed BIN16	Trigger Type	ON Sampling, 2(Sec) GB62320
Same as the script No. 31044 for the base screen 31029			
Script No.	31045	Script Name	Script31045
Comment	29-31_Multi operation prep		
Data Type	Signed BIN16	Trigger Type	ON GB62323
Same as the script No. 31045 for the base screen 31029			
Script No.	31046	Script Name	Script31046
Comment	29-31_During Multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62325
Same as the script No. 31046 for the base screen 31029			
Script No.	31047	Script Name	Script31047
Comment	29-31_End multiple operation		
Data Type	Signed BIN16	Trigger Type	ON GB62326
Same as the script No. 31047 for the base screen 31029			

#### Window screen 30004

Script No.	30100	Script Name	Script30100
Comment	Axis switching		
Data Type	Signed BIN16	Trigger Type	When closing a screen
<pre>// Transfer Axis switching data to each device from dummy device if([b:GB61000] == ON){     [w:GD61000] = [w:GD65000];      if([w:GD65001] == 1){         [b:GB61010] = OFF;     }     if([w:GD65001] == 2){         [b:GB61010] = ON;     } }  [w:GD65000]=0; [w:GD65001]=0; rst([b:GB61000]); rst([b:GB61001]); rst([b:GB61002]);</pre>			

#### Window screen 32001

Script No.	31006	Script Name	Script31006
Comment	Write operation input data		
Data Type	Signed BIN16	Trigger Type	ON GB62003
<pre>// Write the value that were set in the overlap window for the operation data input into the driver // Display it in the operation data screen  // Script No.31006 for the Overlap window screen for the operation data input</pre>			

```

// Start the trigger in case [Setting] button of above mentioned overlap window

// [b:GB62003] : Starting trigger of this script using bit set ([Setting] button)
// [w:GD61011] : Modbus address offset value
// [w:GD61012] : Touch position info of the operation data No. (which row)

// Change the touched row data to reflect the operation data (write data into the driver and change the operation
data screen)
switch ([w:GD61012])
{
    case 0 : [u16:401282[w:GD61011]] = [u16:GD61021];    // Operation mode of line 0
        [s32:401025[w:GD61011]] = [s32:GD61022];    // Position of line 0
        [u32:401153[w:GD61011]] = [u32:GD61024];    // Operating speed of line 0
        [u16:401410[w:GD61011]] = [u16:GD61026];    // Operation function of line 0
        [u32:401793[w:GD61011]] = [u32:GD61027];    // Push current of line 0
        [u32:402049[w:GD61011]] = [u32:GD61029];    // Dwell time of line 0
        [u16:401922[w:GD61011]] = [u16:GD61031];    // Sequential positioning of line 0
        [u32:401537[w:GD61011]] = [u32:GD61032];    // Acceleration of line 0
        [u32:401665[w:GD61011]] = [u32:GD61034];    // Deceleration of line 0
        break;

    case 1 : [u16:401284[w:GD61011]] = [u16:GD61021];    // Operation mode of line 1
        [s32:401027[w:GD61011]] = [s32:GD61022];    // Position of line 1
        [u32:401155[w:GD61011]] = [u32:GD61024];    // Operating speed of line 1
        [u16:401412[w:GD61011]] = [u16:GD61026];    // Operation function of line 1
        [u32:401795[w:GD61011]] = [u32:GD61027];    // Push current of line 1
        [u32:402051[w:GD61011]] = [u32:GD61029];    // Dwell time of line 1
        [u16:401924[w:GD61011]] = [u16:GD61031];    // Sequential positioning of line 1
        [u32:401539[w:GD61011]] = [u32:GD61032];    // Acceleration of line 1
        [u32:401667[w:GD61011]] = [u32:GD61034];    // Deceleration of line 1
        break;

    case 2 : [u16:401286[w:GD61011]] = [u16:GD61021];    // Operation mode of line 2
        [s32:401029[w:GD61011]] = [s32:GD61022];    // Position of line 2
        [u32:401157[w:GD61011]] = [u32:GD61024];    // Operating speed of line 2
        [u16:401414[w:GD61011]] = [u16:GD61026];    // Operation function of line 2
        [u32:401797[w:GD61011]] = [u32:GD61027];    // Push current of line 2
        [u32:402053[w:GD61011]] = [u32:GD61029];    // Dwell time of line 2
        [u16:401926[w:GD61011]] = [u16:GD61031];    // Sequential positioning of line 2
        [u32:401541[w:GD61011]] = [u32:GD61032];    // Acceleration of line 2
        [u32:401669[w:GD61011]] = [u32:GD61034];    // Deceleration of line 2
        break;

    case 3 : [u16:401288[w:GD61011]] = [u16:GD61021];    // Operation mode of line 3
        [s32:401031[w:GD61011]] = [s32:GD61022];    // Position of line 3
        [u32:401159[w:GD61011]] = [u32:GD61024];    // Operating speed of line 3
        [u16:401416[w:GD61011]] = [u16:GD61026];    // Operation function of line 3
        [u32:401799[w:GD61011]] = [u32:GD61027];    // Push current of line 3
        [u32:402055[w:GD61011]] = [u32:GD61029];    // Dwell time of line 3
        [u16:401928[w:GD61011]] = [u16:GD61031];    // Sequential positioning of line 3
        [u32:401543[w:GD61011]] = [u32:GD61032];    // Acceleration of line 3
        [u32:401671[w:GD61011]] = [u32:GD61034];    // Deceleration of line 3
        break;

    case 4 : [u16:401290[w:GD61011]] = [u16:GD61021];    // Operation mode of line 4
        [s32:401033[w:GD61011]] = [s32:GD61022];    // Position of line 4
        [u32:401161[w:GD61011]] = [u32:GD61024];    // Operating speed of line 4
        [u16:401418[w:GD61011]] = [u16:GD61026];    // Operation function of line 4
        [u32:401801[w:GD61011]] = [u32:GD61027];    // Push current of line 4
        [u32:402057[w:GD61011]] = [u32:GD61029];    // Dwell time of line 4
        [u16:401930[w:GD61011]] = [u16:GD61031];    // Sequential positioning of line 4
        [u32:401545[w:GD61011]] = [u32:GD61032];    // Acceleration of line 4
        [u32:401673[w:GD61011]] = [u32:GD61034];    // Deceleration of line 4
        break;
}

```

```

case 5 : [u16:401292[w:GD61011]] = [u16:GD61021];    // Operation mode of line 5
[s32:401035[w:GD61011]] = [s32:GD61022];    // Position of line 5
[u32:401163[w:GD61011]] = [u32:GD61024];    // Operating speed of line 5
[u16:401420[w:GD61011]] = [u16:GD61026];    // Operation function of line 5
[u32:401803[w:GD61011]] = [u32:GD61027];    // Push current of line 5
[u32:402059[w:GD61011]] = [u32:GD61029];    // Dwell time of line 5
[u16:401932[w:GD61011]] = [u16:GD61031];    // Sequential positioning of line 5
[u32:401547[w:GD61011]] = [u32:GD61032];    // Acceleration of line 5
[u32:401675[w:GD61011]] = [u32:GD61034];    // Deceleration of line 5
break;

case 6 : [u16:401294[w:GD61011]] = [u16:GD61021];    // Operation mode of line 6
[s32:401037[w:GD61011]] = [s32:GD61022];    // Position of line 6
[u32:401165[w:GD61011]] = [u32:GD61024];    // Operating speed of line 6
[u16:401422[w:GD61011]] = [u16:GD61026];    // Operation function of line 6
[u32:401805[w:GD61011]] = [u32:GD61027];    // Push current of line 6
[u32:402061[w:GD61011]] = [u32:GD61029];    // Dwell time of line 6
[u16:401934[w:GD61011]] = [u16:GD61031];    // Sequential positioning of line 6
[u32:401549[w:GD61011]] = [u32:GD61032];    // Acceleration of line 6
[u32:401677[w:GD61011]] = [u32:GD61034];    // Deceleration of line 6
break;

case 7 : [u16:401296[w:GD61011]] = [u16:GD61021];    // Operation mode of line 7
[s32:401039[w:GD61011]] = [s32:GD61022];    // Position of line 7
[u32:401167[w:GD61011]] = [u32:GD61024];    // Operating speed of line 7
[u16:401424[w:GD61011]] = [u16:GD61026];    // Operation function of line 7
[u32:401807[w:GD61011]] = [u32:GD61027];    // Push current of line 7
[u32:402063[w:GD61011]] = [u32:GD61029];    // Dwell time of line 7
[u16:401936[w:GD61011]] = [u16:GD61031];    // Sequential positioning of line 7
[u32:401551[w:GD61011]] = [u32:GD61032];    // Acceleration of line 7
[u32:401679[w:GD61011]] = [u32:GD61034];    // Deceleration of line 7
break;
}

[b:GB62003] = 0;    // End the starting trigger of this script
[w:GD60004] = 0;    // Close overlap window

```

### Window screen 32004 to 32006

Script No.	31007	Script Name	Script31007
Comment	Write_IN_OUT function selection		
Data Type	Signed BIN16	Trigger Type	ON GB62004
<pre> // Write the function selection of IN, OUT, NET-IN, NET-OUT into the drivers  // IN input function selection, NET-IN input function selection, OUT/NET-OUT output function selection // Each Overlap window script No.31007 // Start the script in case each [Setting] button (GB32004) is ON  // [b:GB62004] : Starting trigger of this script using bit set ([Setting] button) // [b:GB62005] : Bit to distinguish between OUT and NET-OUT (1:OUT,0:NET-OUT) // [u16:GD61040] : Signal number of IN input // [u16:GD61041] : Signal number of OUT output // [u16:GD61042] : Signal number of NET-IN input // [u16:GD61043] : Signal number of NET-OUT output  // [u16:GD61044] : Temporary device of IN input function selection // [u16:GD61046] : Temporary save device of OUT/NET-OUT output function selection // [u16:GD61048] : Temporary save device of NET-IN input function selection  // When to select IN input function if([w:GD60004] == 32004){    // In case IN input function selection window is displayed     switch([u16:GD61040]){         case 0: [u16:404354] = [u16:GD61044];    // Setting for IN0 input function selection </pre>			

```

        break;
    case 1: [u16:404356] = [u16:GD61044];    // Setting for IN1 input function selection
        break;
    case 2: [u16:404358] = [u16:GD61044];    // Setting for IN2 input function selection
        break;
    case 3: [u16:404360] = [u16:GD61044];    // Setting for IN3 input function selection
        break;
    case 4: [u16:404362] = [u16:GD61044];    // Setting for IN4 input function selection
        break;
    case 5: [u16:404364] = [u16:GD61044];    // Setting for IN5 input function selection
        break;
    case 6: [u16:404366] = [u16:GD61044];    // Setting for IN6 input function selection
        break;
    case 7: [u16:404368] = [u16:GD61044];    // Setting for IN7 input function selection
        break;
    }
}

// When to select NET-IN input function
if([w:GD60004] == 32005){    // In case NET-IN input function selection window is displayed
    switch([u16:GD61042]){
        case 0 : [u16:404450] = [u16:GD61048]; // Setting for NET-IN0 input function selection
            break;
        case 1 : [u16:404452] = [u16:GD61048]; // Setting for NET-IN1 input function selection
            break;
        case 2 : [u16:404454] = [u16:GD61048]; // Setting for NET-IN2 input function selection
            break;
        case 3 : [u16:404456] = [u16:GD61048]; // Setting for NET-IN3 input function selection
            break;
        case 4 : [u16:404458] = [u16:GD61048]; // Setting for NET-IN4 input function selection
            break;
        case 5 : [u16:404460] = [u16:GD61048]; // Setting for NET-IN5 input function selection
            break;
        case 6 : [u16:404462] = [u16:GD61048]; // Setting for NET-IN6 input function selection
            break;
        case 7 : [u16:404464] = [u16:GD61048]; // Setting for NET-IN7 input function selection
            break;
        case 8 : [u16:404466] = [u16:GD61048]; // Setting for NET-IN8 input function selection
            break;
        case 9 : [u16:404468] = [u16:GD61048]; // Setting for NET-IN9 input function selection
            break;
        case 10: [u16:404470] = [u16:GD61048]; // Setting for NET-IN10 input function selection
            break;
        case 11: [u16:404472] = [u16:GD61048]; // Setting for NET-IN11 input function selection
            break;
        case 12: [u16:404474] = [u16:GD61048]; // Setting for NET-IN12 input function selection
            break;
        case 13: [u16:404476] = [u16:GD61048]; // Setting for NET-IN13 input function selection
            break;
        case 14: [u16:404478] = [u16:GD61048]; // Setting for NET-IN14 input function selection
            break;
        case 15: [u16:404480] = [u16:GD61048]; // Setting for NET-IN15 input function selection
            break;
    }
}

// When to select OUT/NET-OUT output function
if([w:GD60004] == 32006){    // In case OUT/NET-OUT output function selection window is displayed
    if([b:GB62005] == 1){    // When to select OUT output function
        switch([u16:GD61041]){
            case 0: [u16:404418] = [u16:GD61046];    // Setting for OUT0 output function selection
                break;
            case 1: [u16:404420] = [u16:GD61046];    // Setting for OUT1 output function selection
                break;
        }
    }
}

```



```

    case 2: [u16:404422] = [u16:GD61046];    // Setting for OUT2 output function selection
        break;
    case 3: [u16:404424] = [u16:GD61046];    // Setting for OUT3 output function selection
        break;
    case 4: [u16:404426] = [u16:GD61046];    // Setting for OUT4 output function selection
        break;
    case 5: [u16:404428] = [u16:GD61046];    // Setting for OUT5 output function selection
        break;
}
} else {                                     // When to select NET-OUT output function
    switch([u16:GD61043]){
        case 0 : [u16:404482] = [u16:GD61046]; // Setting for NET-OUT0 output function selection
            break;
        case 1 : [u16:404484] = [u16:GD61046]; // Setting for NET-OUT1 output function selection
            break;
        case 2 : [u16:404486] = [u16:GD61046]; // Setting for NET-OUT2 output function selection
            break;
        case 3 : [u16:404488] = [u16:GD61046]; // Setting for NET-OUT3 output function selection
            break;
        case 4 : [u16:404490] = [u16:GD61046]; // Setting for NET-OUT4 output function selection
            break;
        case 5 : [u16:404492] = [u16:GD61046]; // Setting for NET-OUT5 output function selection
            break;
        case 6 : [u16:404494] = [u16:GD61046]; // Setting for NET-OUT6 output function selection
            break;
        case 7 : [u16:404496] = [u16:GD61046]; // Setting for NET-OUT7 output function selection
            break;
        case 8 : [u16:404498] = [u16:GD61046]; // Setting for NET-OUT8 output function selection
            break;
        case 9 : [u16:404500] = [u16:GD61046]; // Setting for NET-OUT9 output function selection
            break;
        case 10: [u16:404502] = [u16:GD61046]; // Setting for NET-OUT10 output function selection
            break;
        case 11: [u16:404504] = [u16:GD61046]; // Setting for NET-OUT11 output function selection
            break;
        case 12: [u16:404506] = [u16:GD61046]; // Setting for NET-OUT12 output function selection
            break;
        case 13: [u16:404508] = [u16:GD61046]; // Setting for NET-OUT13 output function selection
            break;
        case 14: [u16:404510] = [u16:GD61046]; // Setting for NET-OUT14 output function selection
            break;
        case 15: [u16:404512] = [u16:GD61046]; // Setting for NET-OUT15 output function selection
            break;
    }
}
}

[b:GB62004] = 0; // End the starting trigger of this script
[w:GD60004] = 0; // Close overlap window

```

### 5.6.3 Object script

#### Base screen 30500

Object	Switch	Object ID *1	20035
Script user ID	1		
Data type	Unsigned BIN16	Trigger type	Device Writing
//Prevents exceeding the total number of the document pages. if([u16:GD60081] >= [u16:GD60082]){ [u16:GD60081] = [u16:GD60082] - 1; }			

## Window screen 30003

Object	Numerical display	Object ID *1	10014
Script user ID	1		
Data type	Unsigned BIN16	Trigger type	Rise, GB40
//Obtain Today's Year & Month from Clock Data [w:TMP950] = [w:GS650] & 0xF000; //Obtain Tenths Digit of "Last 2-Digits of Year" from Clock Data for Setting [w:TMP960] = [w:TMP950] >> 12; //Decimal Alignment [w:TMP968] = [w:TMP960] * 10; //BCD->BIN [w:TMP951] = [w:GS650] & 0x0F00; //Obtain Ones Digit of "Last 2-Digits of Year" from Clock Data for Setting [w:TMP961] = [w:TMP951] >> 8; //BCD->BIN [w:TMP973] = 2000 + [w:TMP968] + [w:TMP961]; //Set Year to TMP973 as BIN [w:GD63990] = [w:TMP973]; //Set Year  [w:TMP952] = [w:GS650] & 0x00F0; //Obtain Tenths Digit of Month from Clock Data for Setting [w:TMP962] = [w:TMP952] >> 4; //Decimal Alignment [w:TMP969] = [w:TMP962] * 10; //BCD->BIN [w:TMP953] = [w:GS650] & 0x000F; //Obtain Ones Digit of Month from Clock Data for Setting [w:TMP974] = [w:TMP969] + [w:TMP953]; //Set Month to TMP974 as BIN [w:GD63991] = [w:TMP974]; //Set Month  [w:TMP954] = [w:GS651] & 0xF000; //Obtain Tenths Digit of "Last 2-Digits of Day" from Clock Data for Setting [w:TMP963] = [w:TMP954] >> 12; //Decimal Alignment [w:TMP970] = [w:TMP963] * 10; //BCD->BIN [w:TMP955] = [w:GS651] & 0x0F00; //Obtain Ones Digit of "Last 2-Digits of Day" from Clock Data for Setting [w:TMP964] = [w:TMP955] >> 8; //BCD->BIN [w:TMP975] = [w:TMP970] + [w:TMP964]; //Set Day to TMP975 as BIN [w:GD63992] = [w:TMP975]; //Set Day  [w:TMP956] = [w:GS651] & 0x00F0; //Obtain Tenths Digit of Hour from Clock Data for Setting [w:TMP965] = [w:TMP956] >> 4; //Decimal Alignment [w:TMP971] = [w:TMP965] * 10; //BCD->BIN [w:TMP957] = [w:GS651] & 0x000F; //Obtain Ones Digit of Hour from Clock Data for Setting [w:TMP976] = [w:TMP971] + [w:TMP957]; //Set Hour to TMP976 as BIN [w:GD63993] = [w:TMP976]; //Set Hour  [w:TMP958] = [w:GS652] & 0xF000; //Obtain Tenths Digit of "Last 2-Digits of Minute" from Clock Data for Setting [w:TMP966] = [w:TMP958] >> 12; //Decimal Alignment [w:TMP972] = [w:TMP966] * 10; //BCD->BIN [w:TMP959] = [w:GS652] & 0x0F00; //Obtain Ones Digit of "Last 2-Digits of Minute" from Clock Data for Setting [w:TMP967] = [w:TMP959] >> 8; //BCD->BIN [w:TMP977] = [w:TMP972] + [w:TMP967]; //Set Minute to TMP977 as BIN [w:GD63994] = [w:TMP977]; //Set Minute  [w:TMP993] = [w:GS652] & 0x00F0; //Obtain Tenths Digit of Second from Clock Data for Setting [w:TMP995] = [w:TMP993] >> 4; //Decimal Alignment [w:TMP996] = [w:TMP995] * 10; //BCD->BIN [w:TMP994] = [w:GS652] & 0x000F; //Obtain Ones Digit of Second from Clock Data for Setting [w:TMP978] = [w:TMP996] + [w:TMP994]; //Set Second to TMP978 as BIN [w:GD63995] = [w:TMP978]; //Set Second			
Object	Numerical display	Object ID *1	10015
Script user ID	2		
Data type	Unsigned BIN16	Trigger type	Ordinary
//BIN -> BCD Conversion  [w:TMP979] = [w:GD63990] - 2000; //Last 2-Digits of Year  [w:TMP980] = (([w:TMP979] / 10) << 4) + ([w:TMP979] % 10); //Year BIN -> BCD [w:TMP981] = (([w:GD63991] / 10) << 4) + ([w:GD63991] % 10); //Month BIN -> BCD			

[w:TMP982] = (([w:GD63992] / 10) << 4) + ([w:GD63992] % 10); //Day BIN -> BCD			
[w:TMP983] = (([w:GD63993] / 10) << 4) + ([w:GD63993] % 10); //Hour BIN -> BCD			
[w:TMP984] = (([w:GD63994] / 10) << 4) + ([w:GD63994] % 10); //Minute BIN -> BCD			
[w:TMP985] = (([w:GD63995] / 10) << 4) + ([w:GD63995] % 10); //Second BIN -> BCD			
Object	Numerical display	Object ID *1	10016
Script user ID	3		
Data type	Unsigned BIN16	Trigger type	Ordinary
//Year & Month Setting			
[w:GS513] = ([w:TMP980] << 8) + [w:TMP981]; //Set Year & Month to Change Time Device			
Object	Numerical display	Object ID *1	10017
Script user ID	4		
Data type	Unsigned BIN16	Trigger type	Ordinary
//Date & Time Setting			
[w:GS514] = ([w:TMP982] << 8) + [w:TMP983]; //Set Date & Time to Change Time Device			
Object	Numerical display	Object ID *1	10018
Script user ID	5		
Data type	Unsigned BIN16	Trigger type	Ordinary
//Minute & Second Setting			
[w:GS515] = ([w:TMP984] << 8) + [w:TMP985]; //Set Minute & Second to Change Time Device			
Object	Numerical display	Object ID *1	10019
Script user ID	6		
Data type	Unsigned BIN16	Trigger type	Ordinary
//Day of Week Setting			
[w:TMP986] = [w:GD63990]; //Year (BIN)			
[w:TMP987] = [w:GD63991]; //Month (BIN)			
[w:TMP988] = [w:GD63992]; //Day (BIN)			
if(([w:TMP987] == 1)    ([w:TMP987] == 2)){ //Correction Processing to Calculate January and February as 13th/14th Month			
[w:TMP986] = [w:TMP986] - 1; //Subtract 1 from Year			
[w:TMP987] = [w:TMP987] + 12; //Add 12 to Month			
}			
[w:TMP989] = [w:TMP986]/4; //Create Items Required for Zeller's Congruence			
[w:TMP990] = [w:TMP986]/100; //Create Items Required for Zeller's Congruence			
[w:TMP991] = [w:TMP986]/400; //Create Items Required for Zeller's Congruence			
[w:TMP992] = (13*[w:TMP987]+8)/5; //Create Items Required for Zeller's Congruence			
//Calculate Day of Week Using Zeller's Congruence and Set the Day to Change Time Device			
[w:GS516] = ([w:TMP986]+[w:TMP989]-[w:TMP990]+[w:TMP991]+[w:TMP992]+[w:TMP988])%7;			

\*1 The Object ID might be changed when a screen is utilized.

## 6. MANUAL DISPLAY

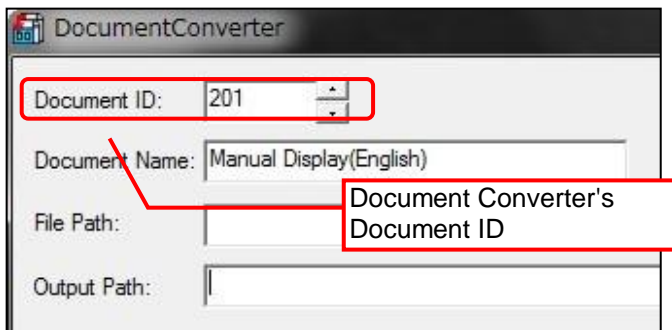
Manuals can be displayed using the document display function. For more details about the document display function, please refer to the "GT Designer3 (GOT2000) Help". Please note that the document display function does not support language switching. Therefore, in the sample screens, the language of document is switched by switching the document (Document ID) specified for a display language.

### 6.1 Preparing Document Data for Manual Display

Example Displaying a English manual (document) for Manual Display on the base screen B-30500

(1) Convert the manual (Word or Excel, etc.) to be displayed into the document data (JPEG file) that can be used with the document display function by using Document Converter. Set the Document Converter's [Document ID] to 201.

\*For details of the relation between Document ID and Display language, please refer to the table below.

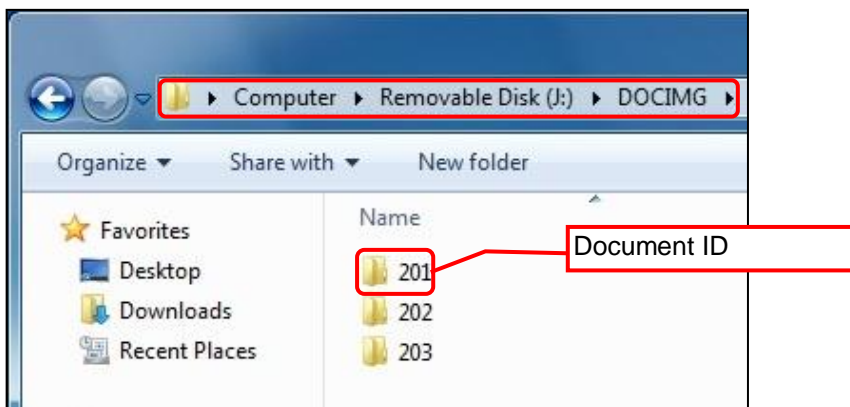


Column No. of the comment group No	Language	Document ID
1	English	201
2	Japanese	202
3	Chinese (Simplified)	203

\*Please use Document Converter 2.09k or later.

The total number pages and pages switches cannot work properly with 2.08 or older versions.

(2) The document data is generated in the 201 folder in the DOCIMG. Save the entire DOCIMG folder into the SD card root directory without changing the folder configuration inside the DOCIMG folder.



SD card folder configuration

Note: In case the total number of pages is 100 or more.

This sample is made with the assumption that the total number of pages is up to 99 pages. If it exceeds 99 pages, please modify the format of numerical input (the number of "#") that displays the total number of pages and the page number of the currently displayed page.

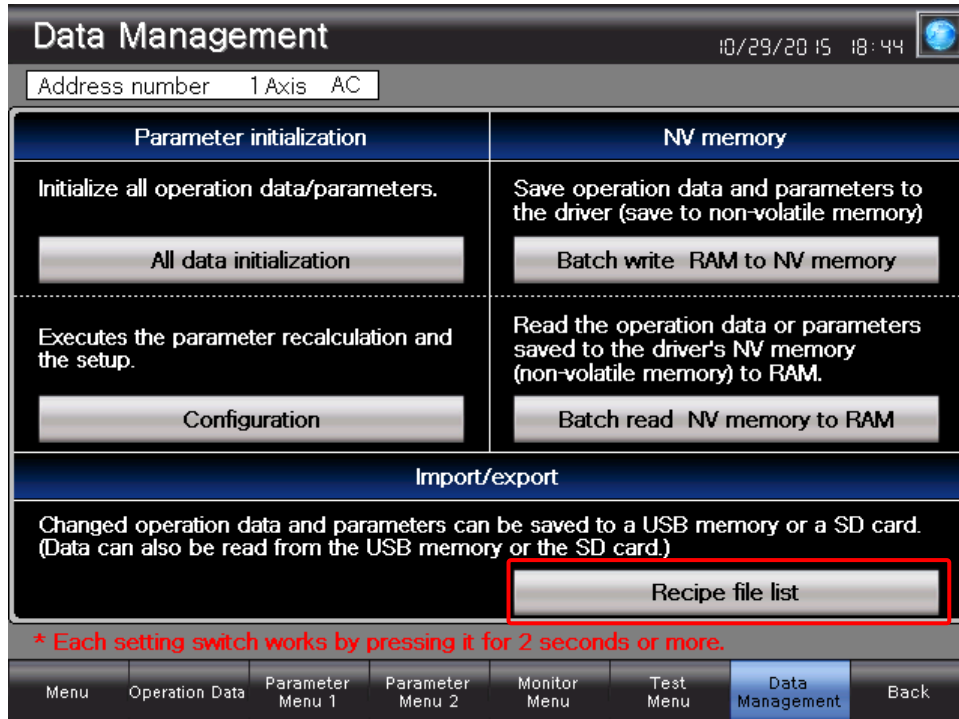
## 7. SAVE/READ OPERATION DATA AND PARAMETER TO/FROM A USB MEMORY OR A CF CARD

Using the recipe file, operation data and parameters can be saved to a USB memory or a CF card and can also be read from the USB memory or the CF card.

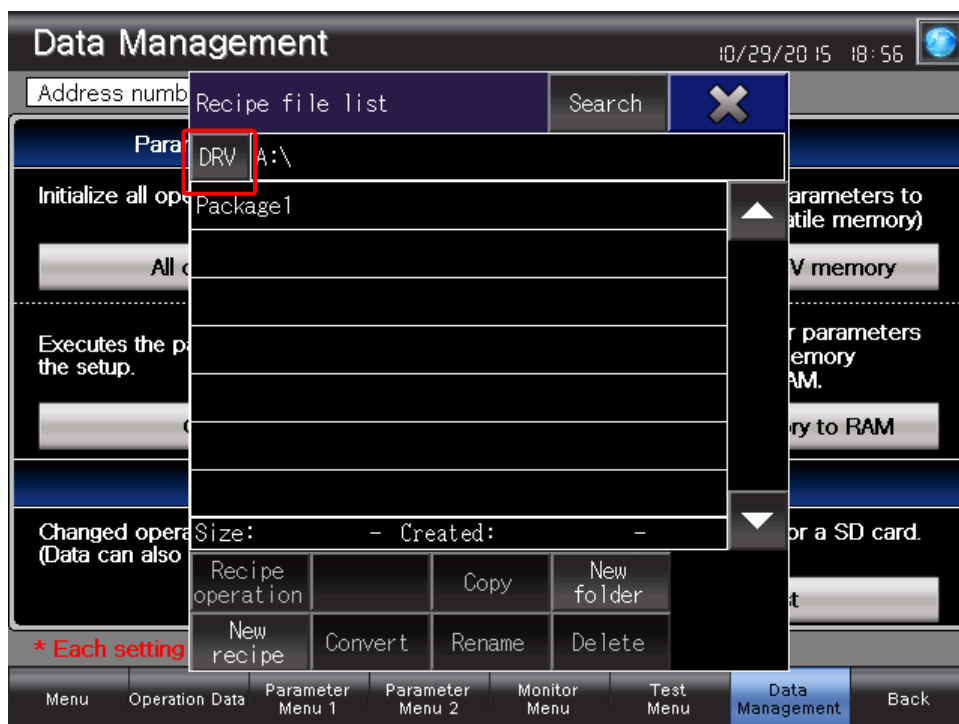
### 7.1 Ways of Saving Operation Data and Parameters to a USB Memory or a CF Card

How to save the data to a USB memory is shown below.

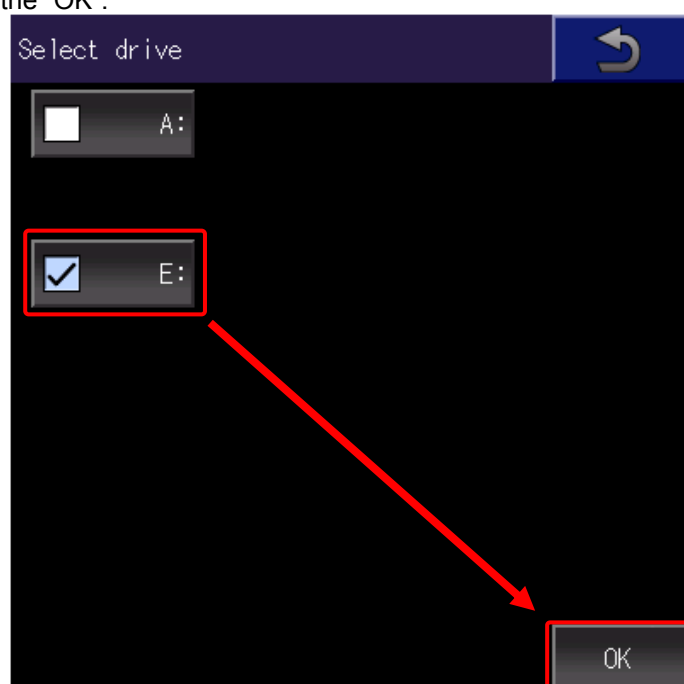
(1) Touch the "Recipe file list" of the [Data management] screen.



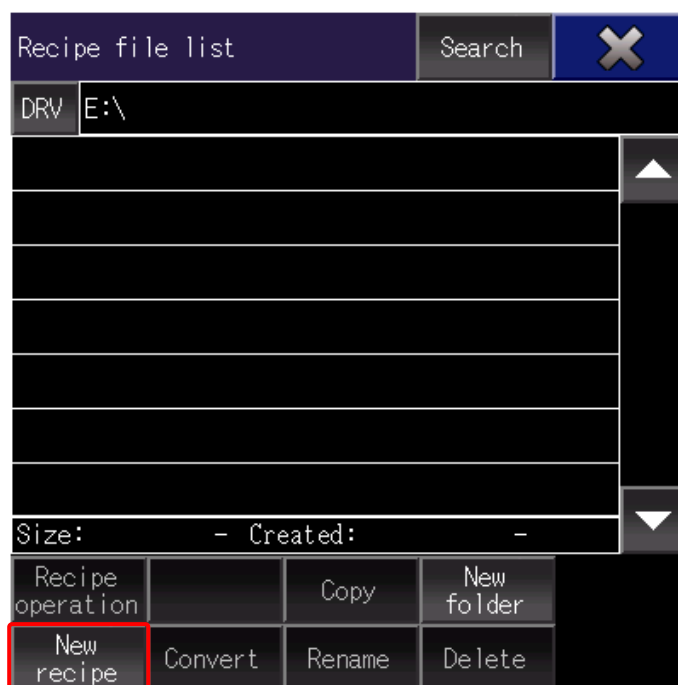
(2) Displays the [Recipe file list] screen. Please touch the "DRV" of the window.



- (3) Displays the [Select drive].  
Please touch the E drive.  
And then please touch the “OK”.



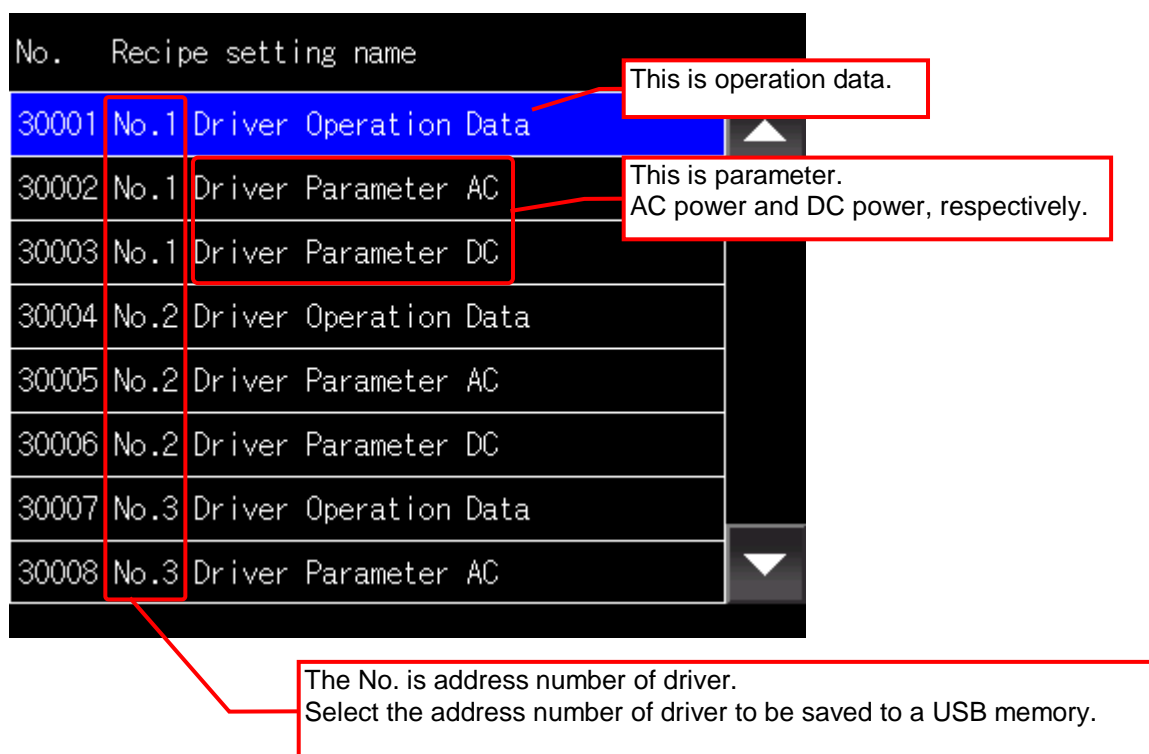
- (4) Please touch the “New recipe”.



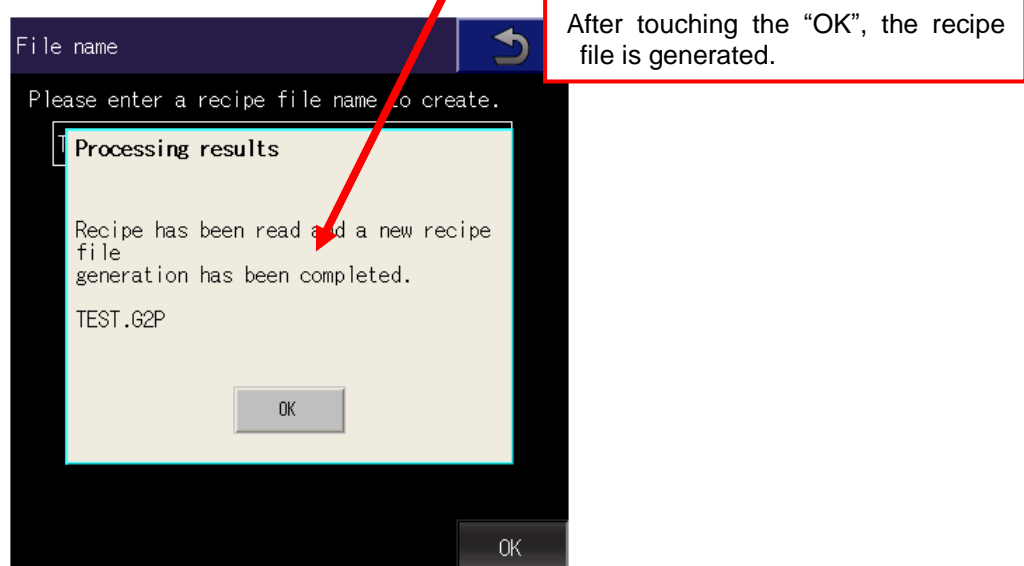
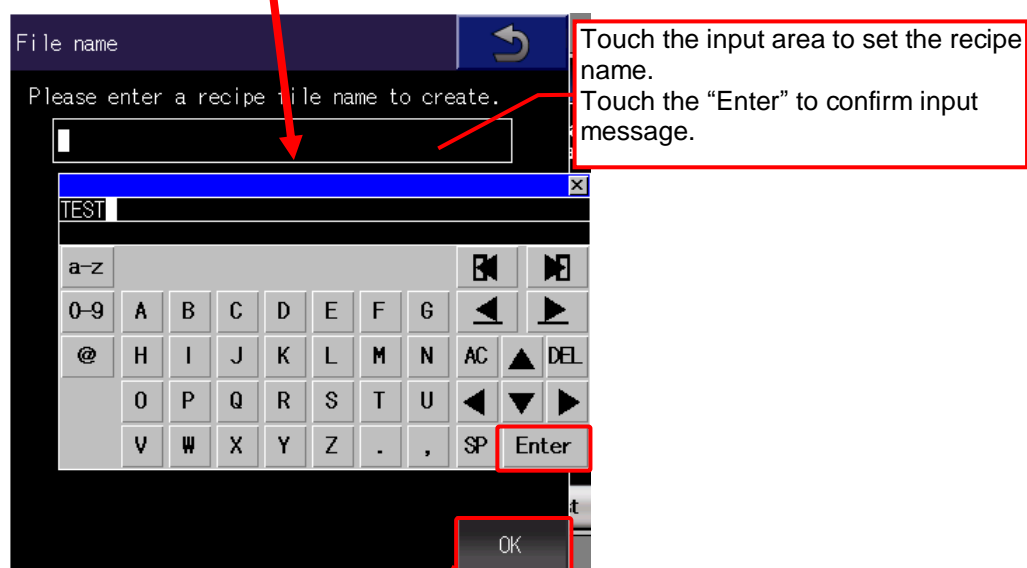
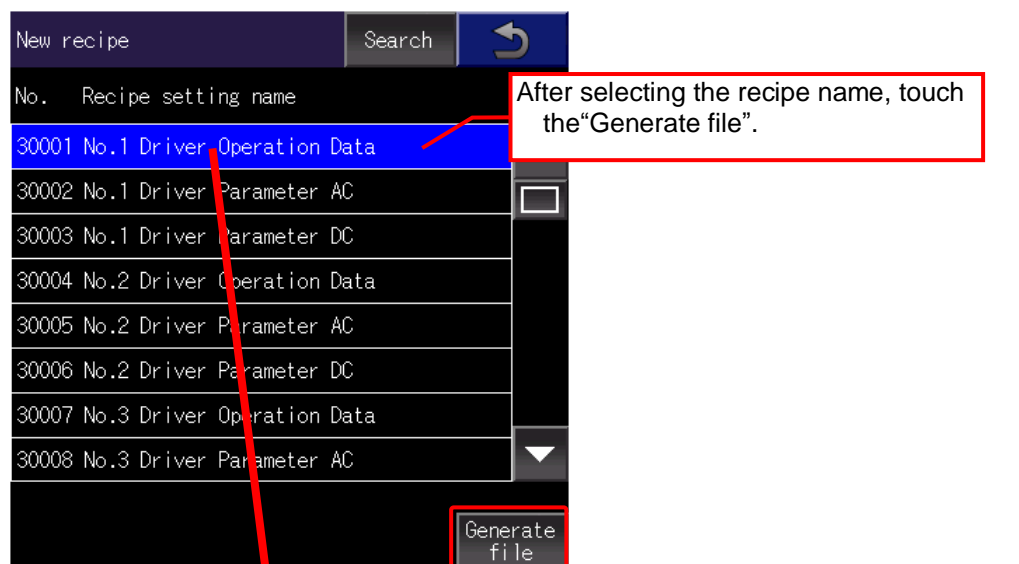
(5) After touching the “New recipe”, recipe setting name (recipe name) list appears. As shown in the figure below:



The meaning of the recipe name is shown below:

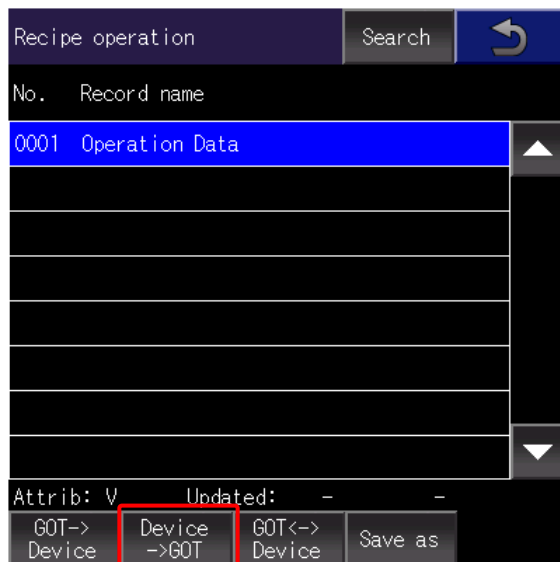


- (6) Select the recipe name (operation data or parameters of each address number of driver) to be saved to a USB memory.  
And then touch the “Generate file”.

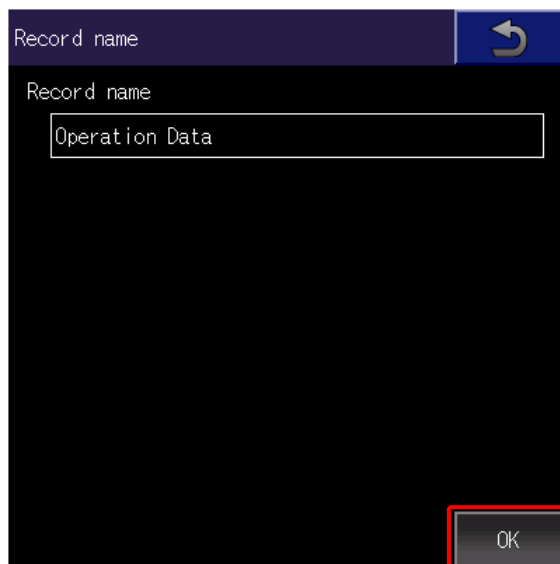




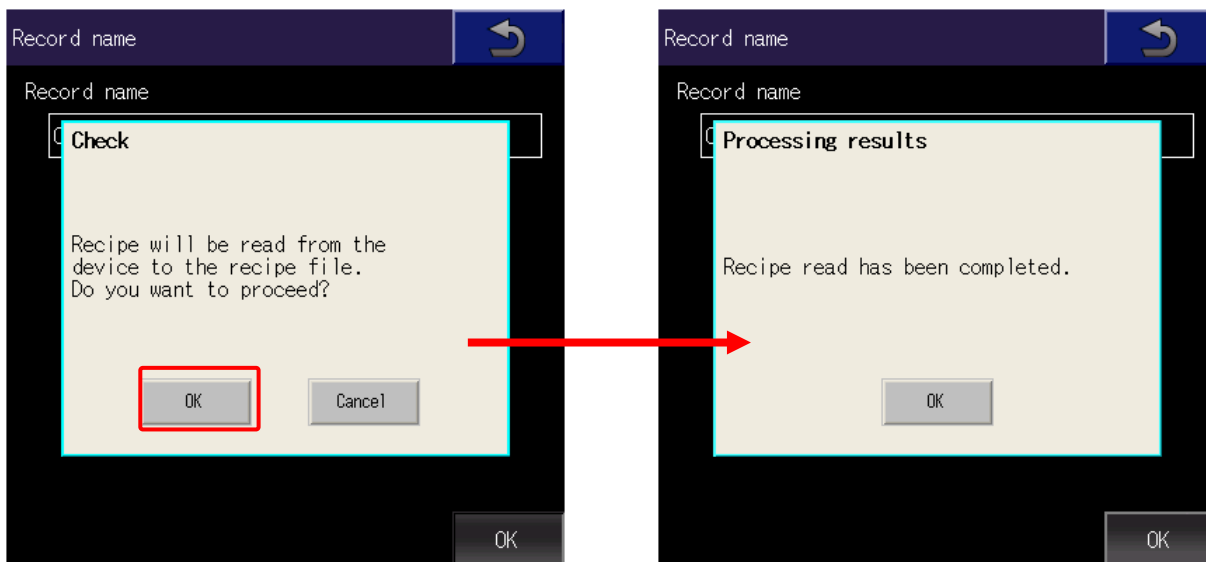
- (7) After generating the recipe file, the record name list appears. As shown in the figure below:  
Here, select the "Operation Data".  
After selecting the "Operation Data", touch the "Device->GOT".



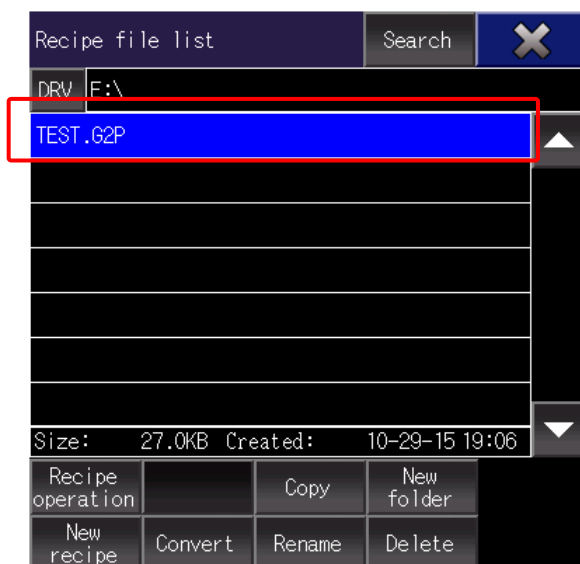
- (8) As shown in the figure below, the record name setting screen appears.  
Without changing the record name, simply touch the "OK".



(9) After touching the “OK”, recipe read is executed.

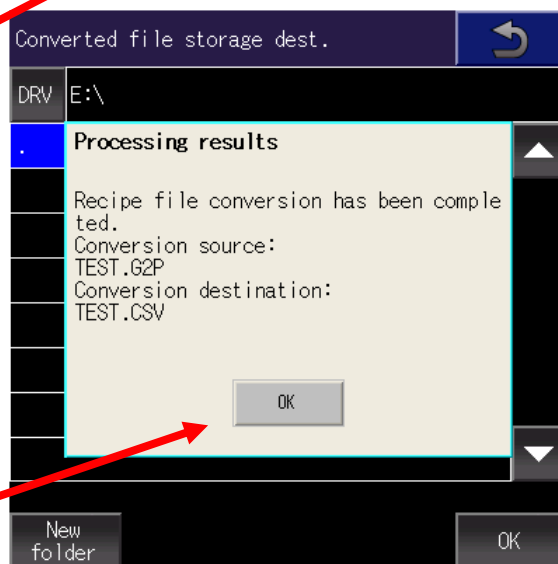
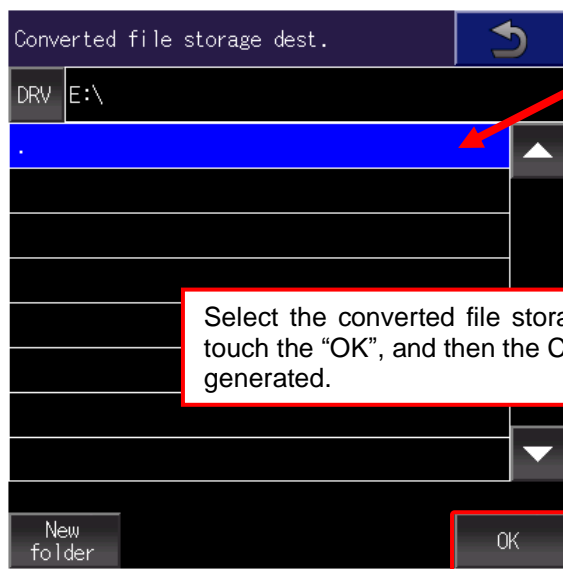
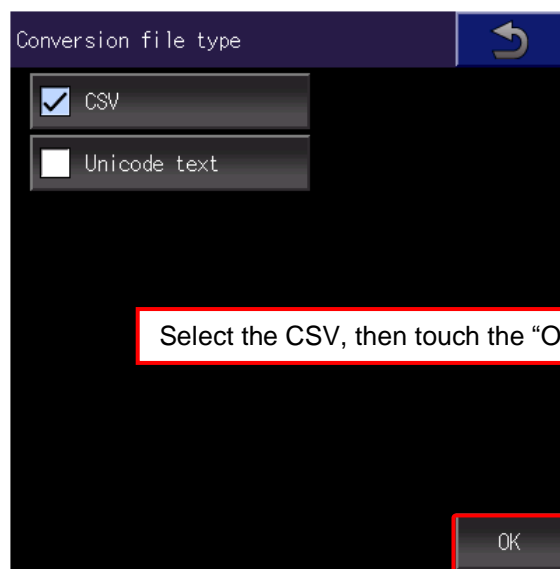
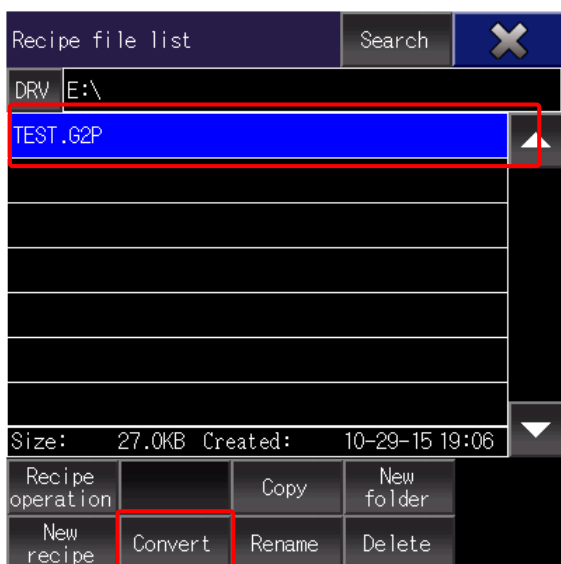


(10) Back to “Recipe file list” screen, in the USB memory, check if there is the recipe file which is generated by reading operation data from the driver.

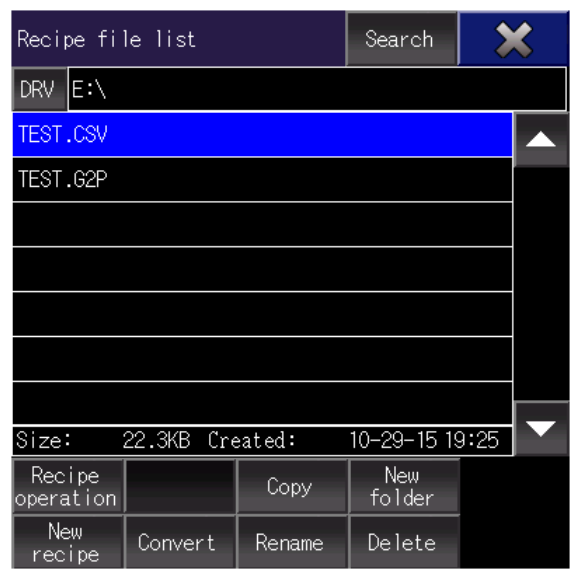


\* Reference: To save the parameters of driver (AC power), select the “Driver Parameter AC” in (6), and then select the “Parameter Data AC” in (7).

- (11) To edit the generated recipe file on computer, convert them into CSV file.  
When the recipe file to be converted is selected, touch the “Convert”.



(12) Through the above operation, the CSV file is generated, as shown in the figure below.

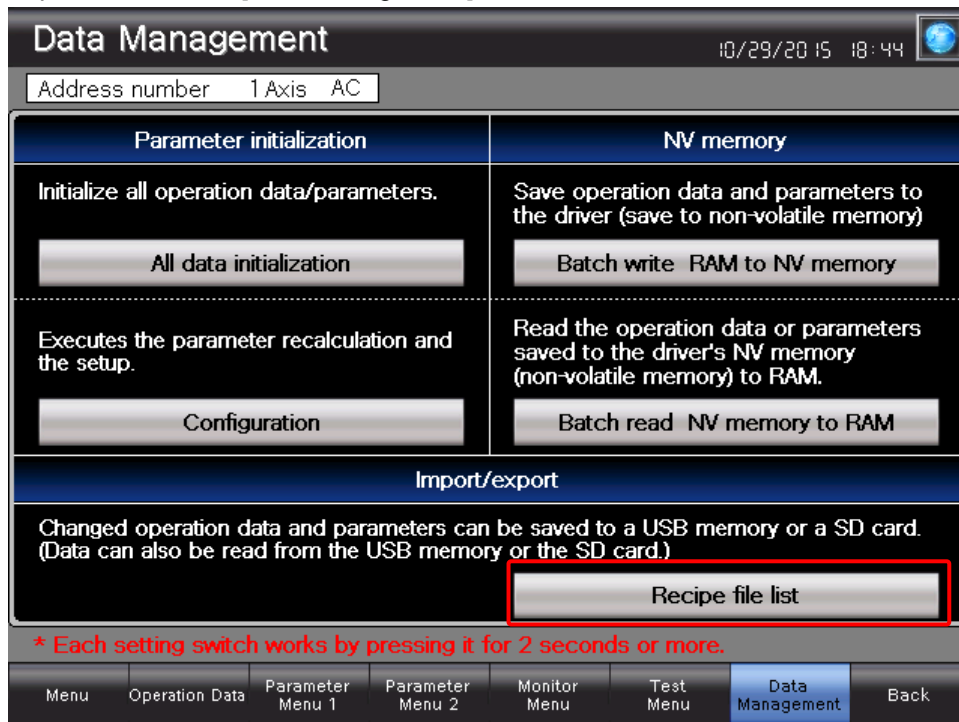


Through the above operation, operation data and parameters of driver can be saved to a USB memory.

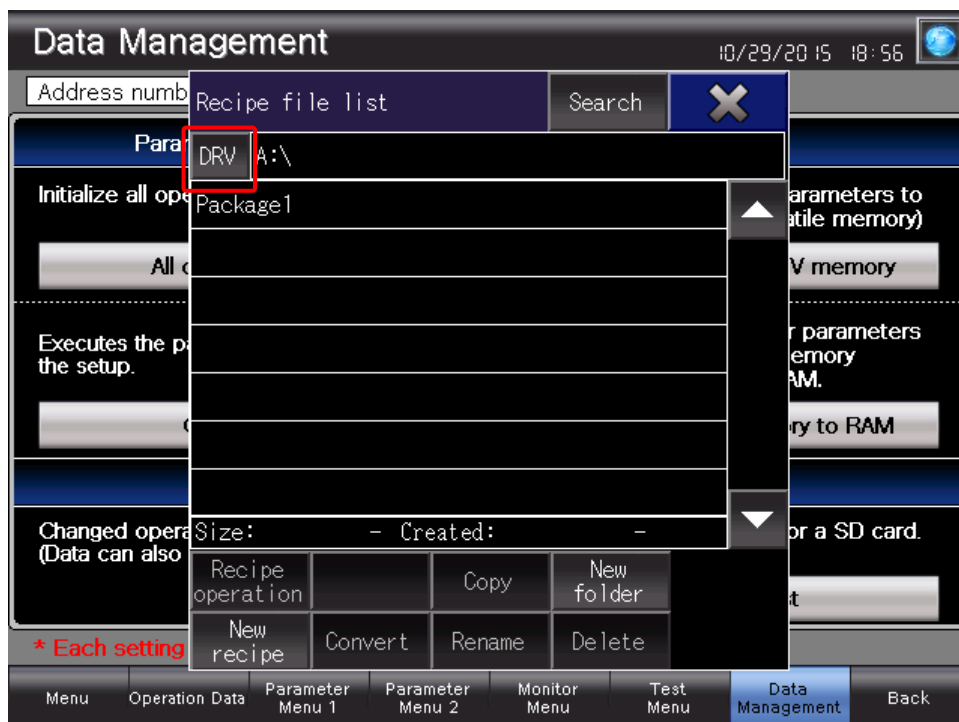
## 7.2 Ways of Reading Operation Data and Parameters from a USB Memory or a CF Card

How to write operation data saved in the USB memory to the driver is shown below.

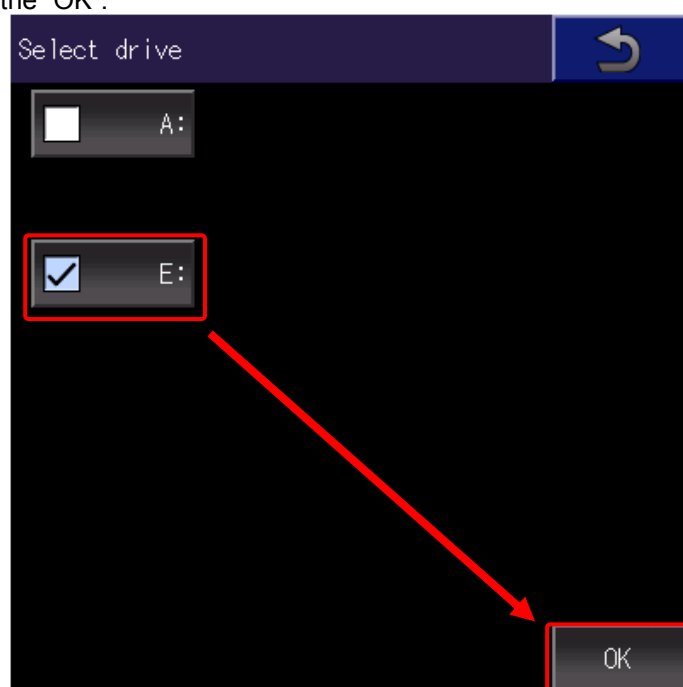
(1) Touch the “Recipe file list” of the [Data management] screen.



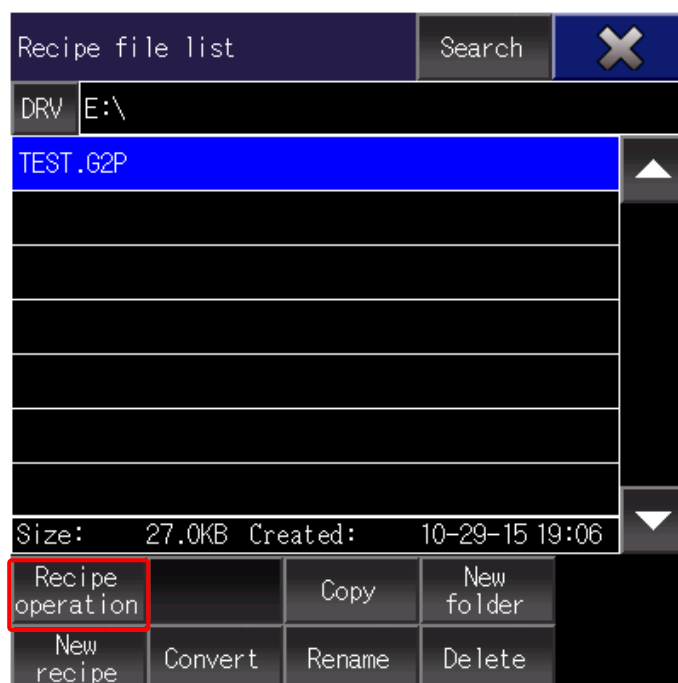
(2) Displays the [Recipe file list] screen. Please touch the “DRV” of the window.



- (3) Displays the [Select drive].  
Please touch the E drive.  
And then please touch the “OK”.



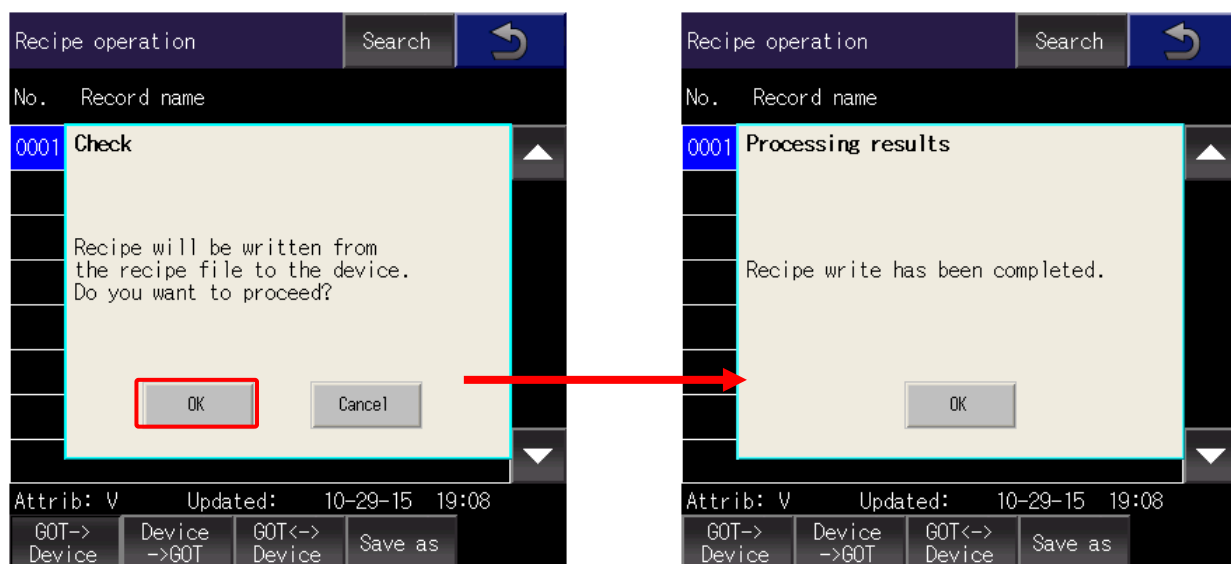
- (4) After choosing the recipe file to be written, please touch the “Recipe operation”.



(5) After selecting the record name, touch the “GOT -> Device”.

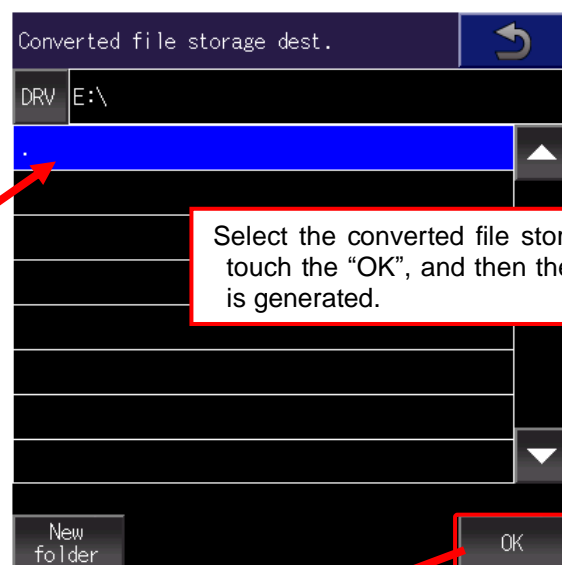
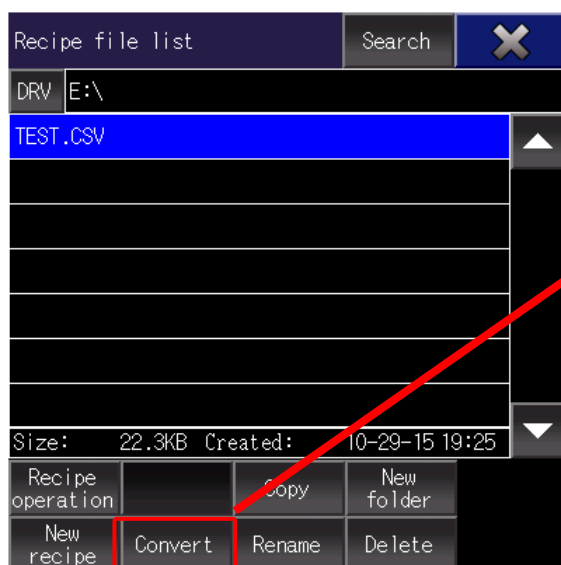


(6) After touching the “OK”, recipe write is executed.

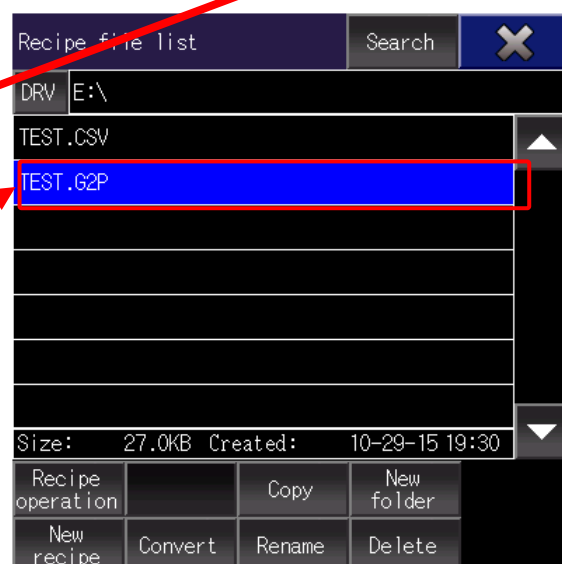
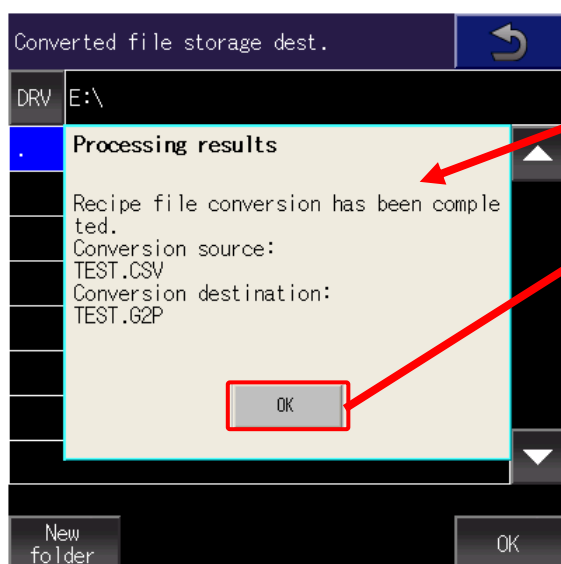


Through the above operation, operation data saved in the USB memory is written to the driver.

- (7) To use the CSV file edited on computer in recipe operation, convert them into G2P file.  
When the CSV file to be converted is selected, touch the “Convert”.



Select the converted file storage dest, touch the “OK”, and then the G2P file is generated.



Through the above operation, the CSV file saved in the USB memory is converted to the G2P file.



## 8. SPECIFICATION OF THE MODBUS ADDRESS OF GOT

GT Designer3 converts the device numbers into decimal format according to the address map of the MODBUS equipment to be used. For more details about the MODBUS address, please refer to "GOT2000 Series Connection Manual (Microcomputer, MODBUS, Products, Peripherals)".

### 8.1 Address Conversion Method

When monitoring the holding register's address "1234H", GT Designer3 displays "4\*\*\*\*\*" since GT Designer3 processes the internal conversion in decimal format as follows: GT Designer3 converts the holding register's address "1234H" to "04660" in decimal format. Then, "+1" is added to this decimal address since the holding register's address on GT Designer3 always starts from "1". Therefore, the holding register's address "1234H" is displayed as "404661" on GT Designer3.

Example: To display the numerical value of the operating speed No.0,

MODBUS register upper address of the operating speed No.0 is "1152" in decimal format.

\* Since the Data Type is set as "Unsigned BIN32", specify upper address of the operating speed No.0.

Actually, the address for the holding register of GOT is "1152+1" = 1153.

And the holding register's address of GOT is assigned to "40000+01153" = 401153.

