

欧立恩拓电机株式会社  
步进电动机组合 α STEP  
高效率 AR 系列 FLEX  
内藏定位功能型 (AC/DC 电源输入)

样本画面说明书

## 关于样本的使用

---

在使用样本画面及其说明书等文件之前，请首先同意以下各项。

- (1) 只有正在使用本公司产品或有意使用本公司产品的用户才能使用。
- (2) 本公司提供的文件的知识产权归属本公司所有。
- (3) 禁止对本公司提供的文件进行窜改、转载、转让、销售。  
但是，可以将部分或全部内容用于用户制作的机器或系统内的本公司产品上。也可以转载、复制、引用、重新排版于本公司用户制作的规格书、设计书、嵌入式产品的使用说明书中。
- (4) 使用本公司提供的文件或从其抽出的数据所造成的任何损失，本公司不予负责。  
请用户自行承担 responsibility。
- (5) 请遵守本公司提供的文件中的使用条件。
- (6) 本公司有权利不经通知修改或删除文件。
- (7) 使用本公司提供的文件时，请务必熟读产品手册以及手册中介绍的相关手册。  
同时请务必充分注意安全事宜，正确使用。

# 目录

---

修订记录.....	5
1. 概要 .....	6
2. 系统构成 .....	6
3. 关于 GOT.....	7
3.1 自动选择的系统应用程序 .....	7
3.2 画面创建软件的连接机器设置 .....	7
3.3 画面创建软件的重叠窗口设置 .....	7
4. 关于驱动器 .....	8
4.1 AC 电源输入驱动器的通信设置.....	8
4.2 DC 电源输入驱动器的通信设置.....	8
4.3 样本的应用对象驱动器 .....	8
5. 画面规格 .....	9
5.1 显示语言 .....	9
5.2 画面一览表/切换 .....	9
5.3 画面说明 .....	15
5.3.1 AC/DC 选择画面(B-30001) .....	15
5.3.2 菜单(B-30002) .....	16
5.3.3 运行数据(B-31002) .....	17
5.3.4 参数菜单 1(B-30003) .....	19
5.3.5 参数 I/O(B-31004) .....	20
5.3.6 参数 电机(B-31005) .....	21
5.3.7 参数 运行(B-31006) .....	22
5.3.8 参数 原点返回(B-31007) .....	23
5.3.9 参数 报警&警告(B-31008) .....	24
5.3.10 参数菜单 2(B-30004) .....	25
5.3.11 参数 坐标(B-31010) .....	26
5.3.12 参数 通用(B-31011) .....	27
5.3.13 参数 I/O 功能(B-31012) .....	28
5.3.14 参数 I/O 功能 RS-485(B-31013) .....	29
5.3.15 参数 通讯(B-31014) .....	30
5.3.16 监视菜单(B-30005) .....	31
5.3.17 监视 状态(B-31015) .....	32
5.3.18 监视 I/O 监视(B-31016) .....	33
5.3.19 监视 I/O RS-485(B-31017) .....	34
5.3.20 监视 报警记录(B-31018) .....	35
5.3.21 监视 警告记录(B-31019) .....	36
5.3.22 手册显示(B-30500) .....	37
5.3.23 测试菜单(B-30006) .....	39
5.3.24 测试 I/O 测试(B-31020) .....	40
5.3.25 测试 运行(B-31021) .....	41
5.3.26 测试 多点运行(B-31022~31029) .....	43
5.3.27 系统 数据管理(B-31030) .....	45
5.3.28 报警复位(W-30001) .....	46
5.3.29 语言设置(W-30002) .....	47
5.3.30 时钟设置(W-30003) .....	48
5.3.31 轴切换(W-30004) .....	49
5.3.32 输入运行数据(W-32001) .....	50
5.3.33 STOP 输入停止方法(W-32002) .....	51

5.3.34	原点返回方法(W-32003)	52
5.3.35	IN 输入功能选择(W-32004)	53
5.3.36	NET-IN 输入功能选择(W-32005)	54
5.3.37	OUT/NET-OUT 输出功能选择(W-32006)	55
5.3.38	报警发生中确认(W-32007)	56
5.3.39	电机运行中确认(W-32008)	57
5.4	使用软元件一览表	58
5.5	注释一览表	64
5.6	脚本一览表	66
6.	关于手册显示	196
6.1	手册显示用文件数据的准备	196
7.	关于对 USB 存储器、SD 卡存取运行数据及参数	197
7.1	运行数据及参数保存至 USB 存储器或 SD 卡的方法	197
7.2	从 USB 存储器或 SD 卡读取运行数据及参数的方法	205
8.	关于 GOT 上的 MODBUS 地址的指定方法	209
8.1	地址的替换方法	209

## 修订记录

### 样本画面说明书

修订日期	管理编号*	修订内容
2014/8	BCN-P5999-0417	初版
2015/6	BCN-P5999-0417-2	文件 ID 的软元件指定对应

\* 管理编号记载在右下方。

### 工程数据

修订日期	工程数据	GT Designer3*	修订内容
2014/8	ORIENTAL_AR-MODBUS_V_Ver1_C.GTX	1.117X	初版
2015/6	ORIENTAL_AR-MODBUS_V_Ver2_C.GTX	1.128J	文件 ID 的软元件指定对应

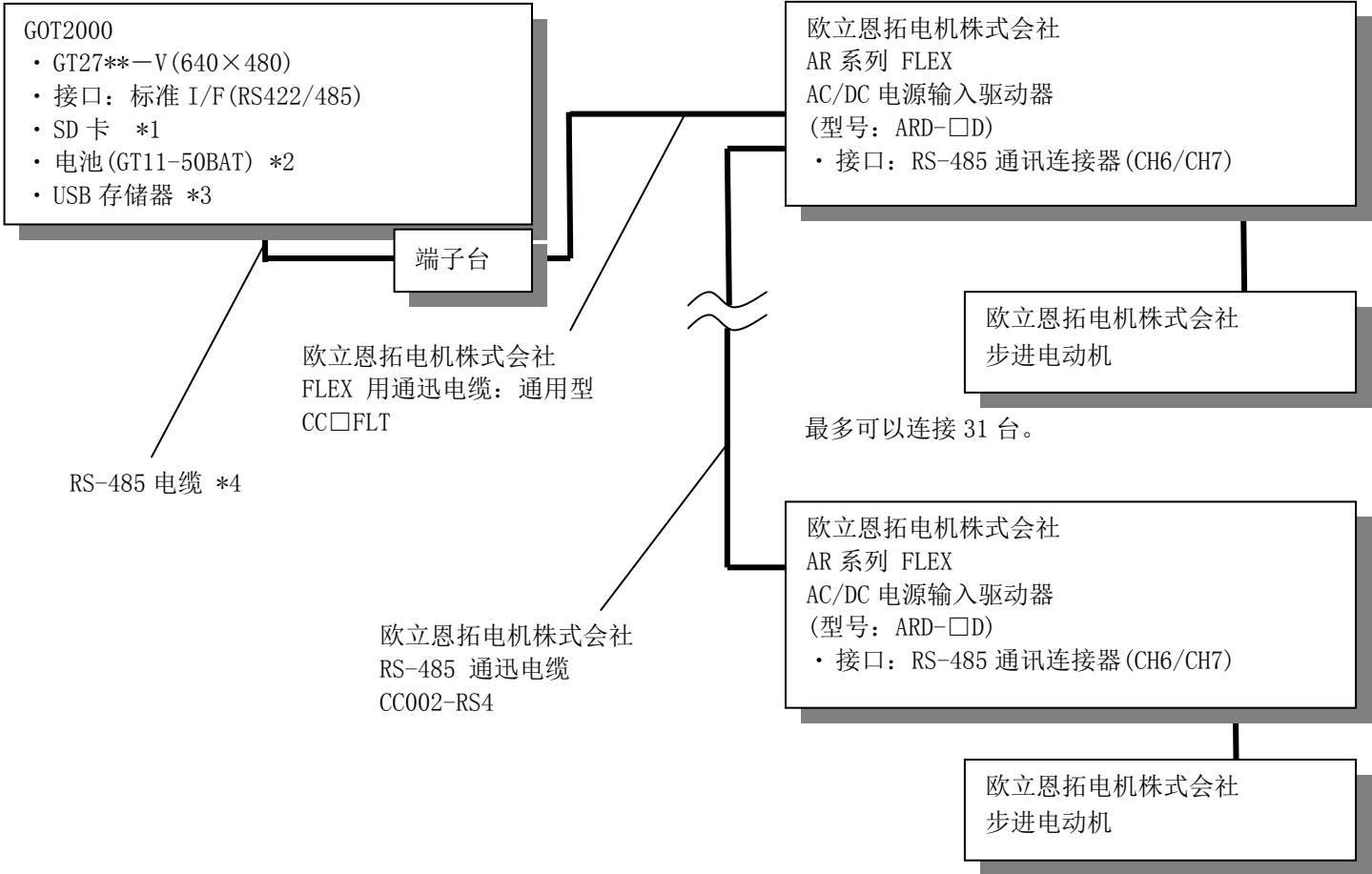
\* 制作工程数据时使用的画面创建软件的版本。打开文件时请使用相同版本或更高版本的画面创建软件。

# 1. 概要

本资料是使用串行 (RS-485) 连接 GOT2000 和欧立恩拓电机株式会社 AR 系列 FLEX 的驱动器 (ARD-□D)，对欧立恩拓电机株式会社步进电动机的当前值和设置值进行监视、更改的样本画面的说明书。

## 2. 系统构成

### 2.1 系统构成



\*1: SD 卡，用于文件显示功能/配方功能。  
\*2: 电池，用于时钟数据的停电保持。(GOT 中标配电池。)  
\*3: USB 存储器，用于配方功能。  
\*4: 关于连接方法的详细内容，请参照「GOT2000 系列 连接手册 (微型计算机/ MODBUS/周边机器连接篇)」。

### 3. 关于 GOT

#### 3.1 自动选择的系统应用程序

种类	系统应用程序名称		
基本功能	基本系统应用程序		
	标准字体	中文(简体)	
通讯驱动程序	MODBUS/RTU		
扩展功能	标准字体		日语
	轮廓字体	黑体	英数假名
			日语汉字
			中文(简体)汉字
	配方操作		
	文件显示		

#### 3.2 画面创建软件的连接机器设置

项目	设置值	备注
波特率(BPS)	115200	
数据长度	8 位	
停止位	1 位	
奇偶性	偶数	
重试次数(次)	2	
通讯超时时间(秒)	3	
本站地址	1	设置监视控制器的轴号码。
发送延迟时间(ms)	8	请设置 8ms 以上
32 位存储顺序	HL 顺序	请设置 HL 顺序
函数代码[0F]	不使用	
函数代码[10]	使用	
线圈读取点数(点)	2000	
输入继电器读取点数(点)	2000	
保持寄存器读取点数(点)	16	最大点数是 16 点。
输入寄存器读取点数(点)	125	
线圈写入点数(点)	800	
保持寄存器写入点数(点)	16	最大点数是 16 点。

#### 3.3 画面创建软件的重叠窗口设置

为了在基本画面切换时关闭窗口画面, 请将[画面切换/窗口]的重叠窗口的[详细设置]中的[切换基本画面的同时关闭窗口]设为有效。

## 4. 关于驱动器

### 4.1 AC 电源输入驱动器的通信设置

本公司动作确认的时候，设置的值如下。

#### (1) 参数设置

项目	设置值	备注
通信停止 bit	1 bit	
通信奇偶	偶数	

#### (2) 驱动器的 DIP 开关、旋转开关设置

项目	设置值	备注
号机设定开关 (ID)	ID=1	更改每个连接驱动器的值
通信速度设定开关 (SW2)	SW2=4	115200bps
功能设定开关 (SW4)	No. 2=ON	ON: 选择 Modbus 协议
终端电阻设定开关 (TERM. )	No. 1、No. 2=ON	仅仅设置最远位置 (终端) 的驱动器的终端电阻为ON

### 4.2 DC 电源输入驱动器的通信设置

本公司动作确认的时候，设置的值如下。

#### (1) 参数设置

项目	设置值	备注
通信停止 bit	1 bit	
通信奇偶	偶数	

#### (2) 驱动器的 DIP 开关、旋转开关设置

项目	设置值	备注
号机设定开关 (SW1)	ID=1	更改每个连接驱动器的值
通信速度设定开关 (SW2)	SW2=4	115200bps
功能设定开关 (SW3 No. 2)	No. 2=ON	ON: 选择 Modbus 协议
终端电阻设定开关 (SW3 No. 4)	No. 4=ON	仅仅设置最远位置 (终端) 的驱动器的终端电阻为ON

### 4.3 样本的应用对象驱动器

样本与以下的生产日期或驱动器版本的驱动器相对应。

#### (1) 生产日期

2014 年 1 月以后

\*生产日期，在驱动器的标签上所记载。

#### (2) 驱动器版本

AC 电源输入驱动器: Ver. 2.00 以后

DC 电源输入驱动器: Ver. 2.01 以后

\*驱动器版本通过数据设定软件 MEXE02 的状态，I/O 监视可以确认。



5. 画面规格

5.1 显示语言

画面可以显示日语/英语/中文(简体)3种语言。如下所示各种语言的字符串，登录在注释组号 497~500 的列号 1~3 中。将列号写入语言切换软件中即可显示与列号相应的语言。

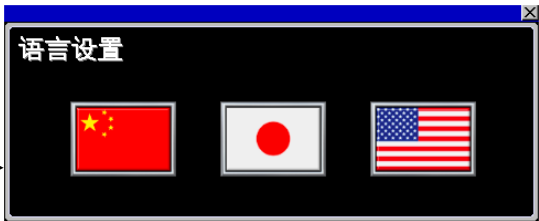
列号	语言
1	中文(简体)
2	日语
3	英语

5.2 画面一览表/切换

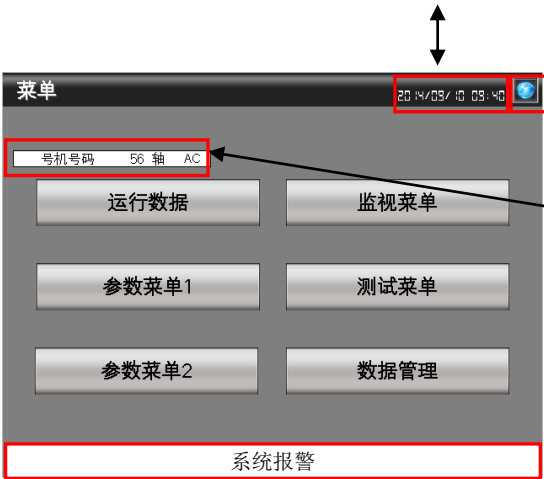
5.2.1 画面一览表/切换(公共)



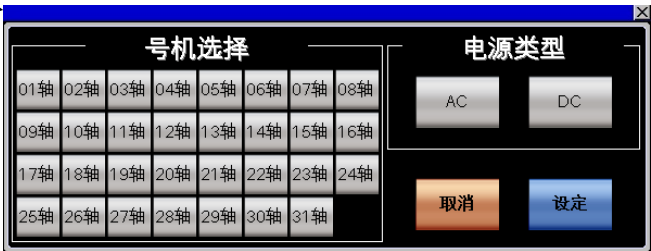
窗口画面 W-30003：时钟设置



窗口画面 W-30002：语言设置



基本画面：全部基本画面



窗口画面 W-30004：轴切换(除 B-30001 显示)

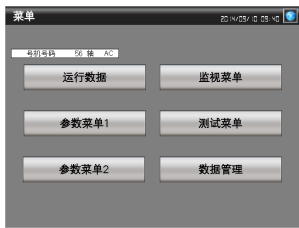


窗口画面 W-30001：报警复位

5.2.2 画面一览表/切换(个别)



基本画面 B-30001：AC/DC 选择画面



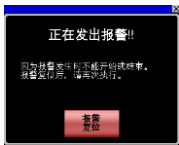
基本画面 B-30002：菜单



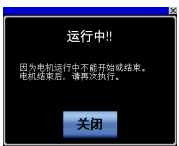
基本画面 B-31002：运行数据



窗口画面 W-32001：  
运行数据输入



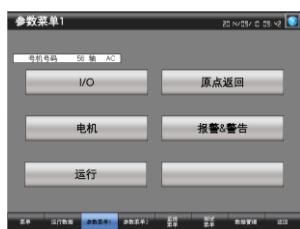
窗口画面 W-32007：  
报警发生中确认



窗口画面 W-32008：  
电机运行中确认

至下一页

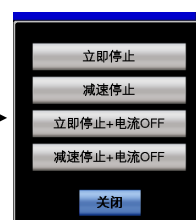
接上一页



基本画面 B-30003:  
参数菜单 1



基本画面 B-31004:  
参数 I/O



窗口画面 W-32002:  
STOP 输入停止方法



基本画面 B-31005:  
参数 电机



基本画面 B-31006:  
参数 运行



基本画面 B-31007:  
参数 原点返回



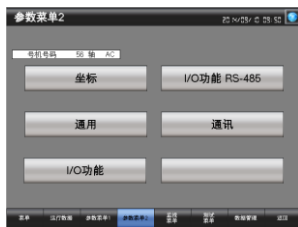
窗口画面 W-32003:  
原点返回方法



基本画面 B-31008:  
参数 报警&警告

至下一页

接上一页



基本画面 B-30004:  
参数菜单 2



基本画面 B-31010:  
参数 坐标



基本画面 B-31011:  
参数 通用



基本画面 B-31012:  
参数 I/O 功能



基本画面 B-31013:  
参数 I/O 功能 RS-485



基本画面 B-31014:  
参数 通讯



窗口画面 W-32004:  
IN 输入功能选择



窗口画面 W-32006:  
OUT/NET-OUT 输出功能选择



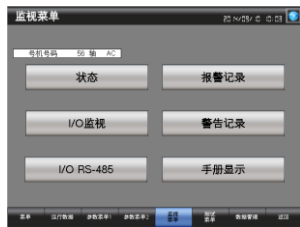
窗口画面 W-32005:  
NET-IN 输入功能选择



窗口画面 W-32006:  
OUT/NET-OUT 输出功能选择

至下一页

接上一页



基本画面 B-30005:  
监视菜单



基本画面 B-31015:  
监视 状态



基本画面 B-31016:  
监视 I/O 监视



基本画面 B-31017:  
监视 I/O RS-485



基本画面 B-31018:  
监视 报警记录



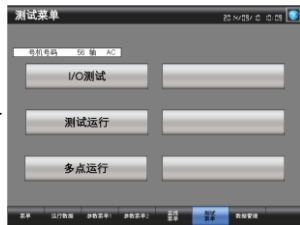
基本画面 B-31019:  
监视 警告记录



基本画面 B-30500:  
手册显示

至下一页

接上一页



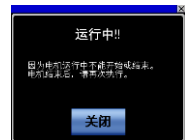
基本画面 B-30006:  
测试菜单



基本画面 B-31020:  
测试 I/O 测试



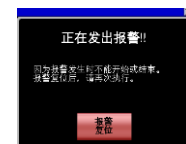
窗口画面 W-32007:  
报警发生中确认



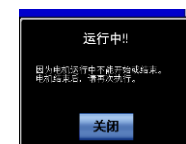
窗口画面 W-32008:  
电机运行中确认



基本画面 B-31021:  
测试 运转



窗口画面 W-32007:  
报警发生中确认



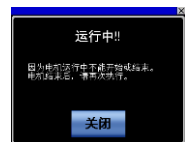
窗口画面 W-32008:  
电机运行中确认



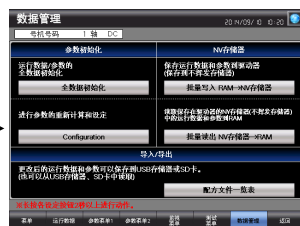
基本画面 B-31022~31029:  
测试 多点运行



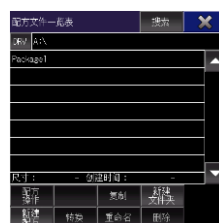
窗口画面 W-32007:  
报警发生中确认



窗口画面 W-32008:  
电机运行中确认



基本画面 B-31030:  
系统 数据管理



系统画面  
配方文件一览表窗口

5.3 画面说明

5.3.1 AC/DC 选择画面(B-30001)

AC/DC选择画面

2014/09/10 09:40

1

号机号码 56 轴

2

AR-AC

AR-DC

3

4

概要

指定 AC/DC 电源。

详细

1. 显示所监视驱动器的号机号码。触摸即可更改号机号码。

2. 指定所监视驱动器的电源类型。

3. 显示当前日期和时间。触摸即显示时钟设置窗口。

4. 显示语言设置窗口。

备注

• 监视多台驱动器时，请务必保证在连接机器设置的本站地址中所设置号机号码的驱动器的存在。此样本中本站地址设置为「1」。关于设置本站地址的详细内容，请参照「GOT2000 系列 连接手册(微型计算机/MODBUS/周边机器连接篇)对应 GT Works3 Version1」。

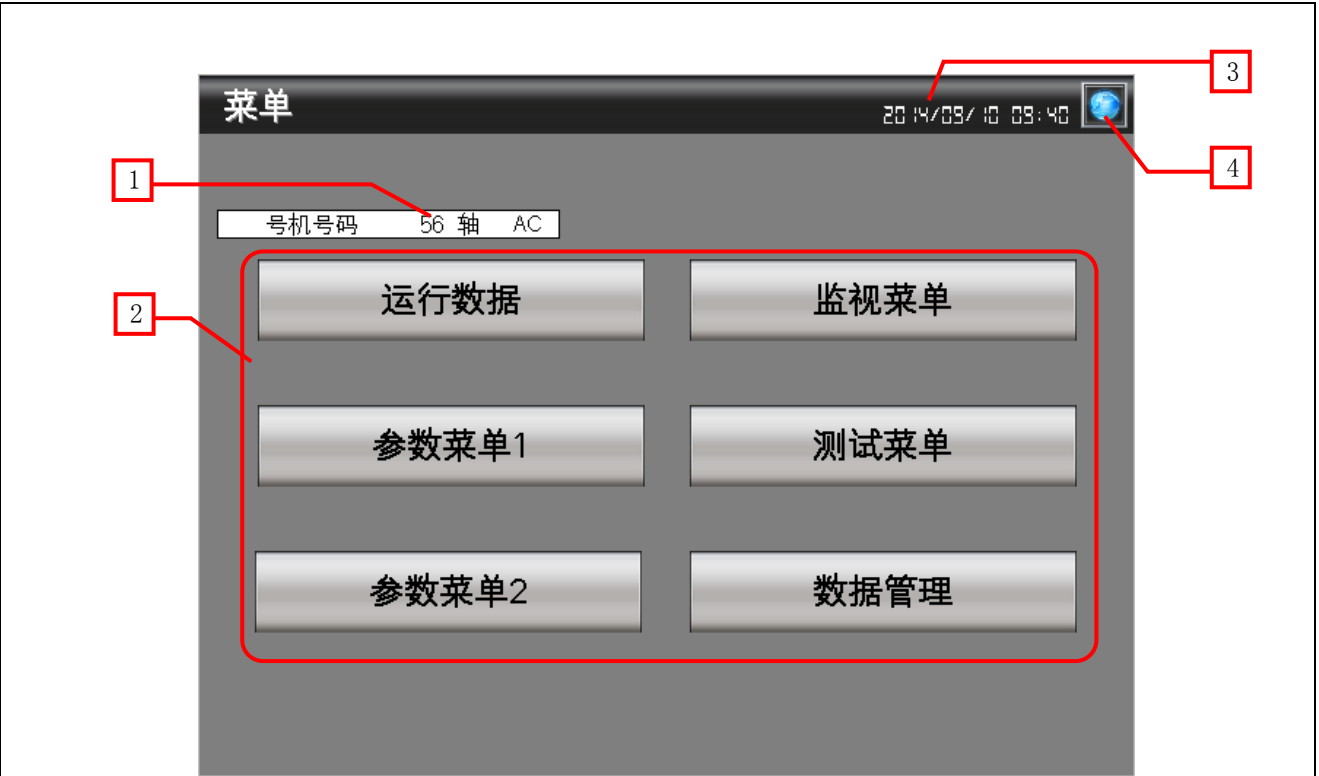
• GOT 启动时，通过工程脚本将号机号码设置为「1」。关于脚本的详细内容，请参照「5.6 脚本一览表」。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

15/209

BCN-P5999-0417-2

5.3.2 菜单(B-30002)



概要

菜单画面。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 切换至各画面。
3. 显示当前日期和时间。触摸即显示时钟设置窗口。
4. 显示语言设置窗口。

备注

- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。









5.3.3 运行数据(B-31002)



概要

显示/更改驱动器的运行数据。并且操作步进电动机。

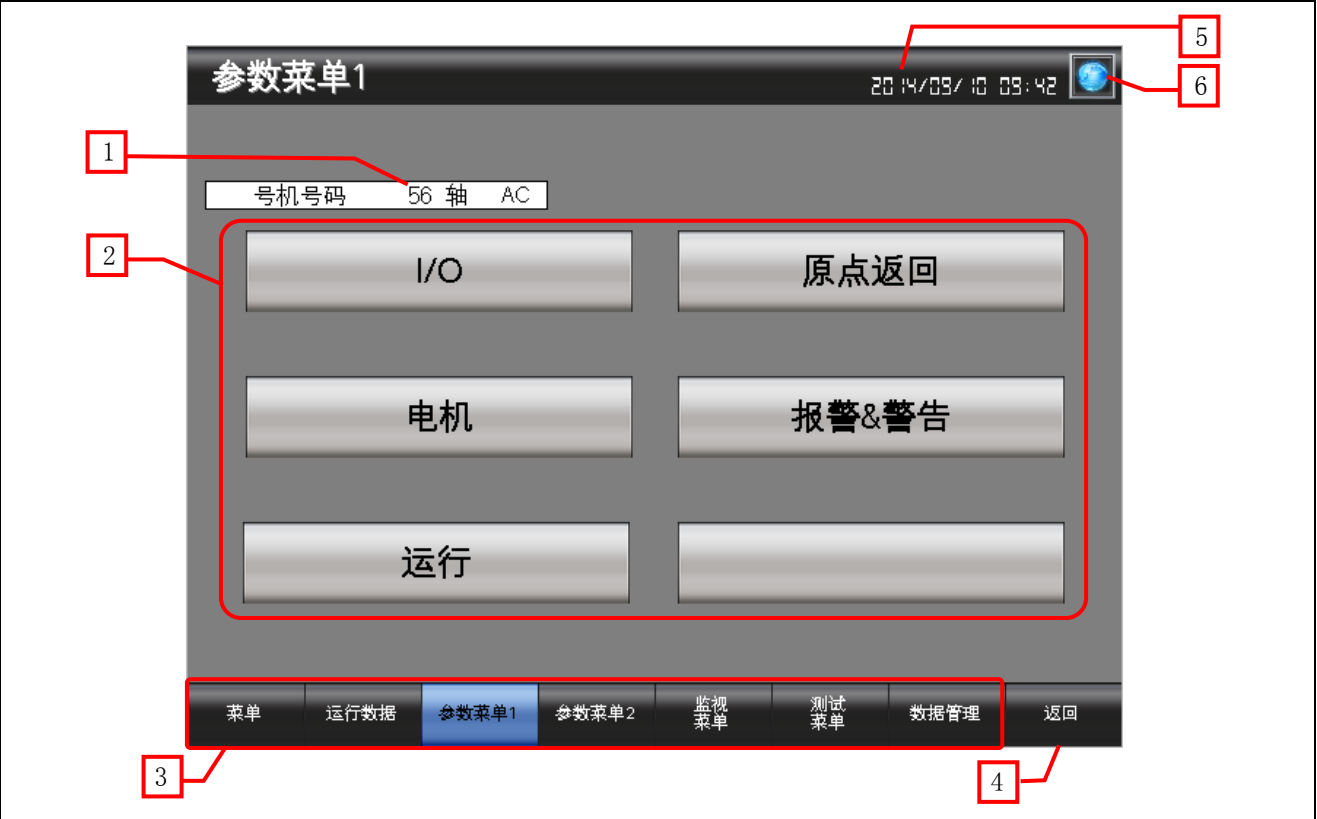
详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 显示运行数据。触摸运行数据即可显示运行数据输入窗口，在此窗口中可以设置相应的运行数据。
3. 显示电机的指令位置。
4. 勾选开始示教运行即可开始示教运行。
  - 最小移动量 : 可设置电机动作的最小移动量。
  - 运行数据 No. : 选择运行数据 No.。
  -   : 持续按压开关期间，连续运行正转/反转。
  -   : 调整电机的位置。电机仅移动最小移动量中所设置的移动量。
  - 停止 : 停止运行中的电机。
  - 定位运行 : 按照运行数据 No. 中所选的运行数据执行定位运行。
  - 原点返回运行 : 开始原点返回运行。
  - 位置确定 : 将电机所在位置反映至运行数据 No. 中所选的位置。运行方式也会变成绝对位置 (ABS)。
  - 位置预置 : 将预置值设置到指令位置中。可以在「参数 坐标」的项目「预置位置」中更改预置值。
5. 滚动运行数据。
  -  : 向上滚动 8 件。
  -  : 向下滚动 8 件。
6. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
7. 切换至上次显示画面。
8. 显示当前日期和时间。触摸即显示时钟设置窗口。
9. 显示语言设置窗口。

#### 备注

- 示教运行开始或结束时，电机的励磁会有一瞬间被切断，因此在用作升降轴等的情况下需要额外注意。而且，电机在瞬间切断励磁后再次励磁，位置会发生偏移。如果要追求位置精度，建议在示教运行结束后立刻执行原点返回运行。
- 在示教运行中，加减速选择参数的设定为独立时，将更改为加速、减速选择的运行数据设定。
- 示教运行中，不能切换到其他画面，也不能更改号机号码。
- 通过画面脚本执行示教运行。关于脚本的详细内容，请参照「5.6 脚本一览表」。
- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.4 参数菜单 1 (B-30003)



概要

参数菜单 1 画面。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 切换至各画面。
3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
4. 切换至上次显示画面。
5. 显示当前日期和时间。触摸即显示时钟设置窗口。
6. 显示语言设置窗口。

备注

- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.5 参数 I/O(B-31004)

1

2

3

4

5

6

I/O

2014/09/10 09:43

号机号码 56 轴 AC

STOP输入停止方法	立即停止	MSO运行No.选择	12
硬件超程	无效	MS1运行No.选择	12
超程动作	立即停止	MS2运行No.选择	12
定位结束输出宽度门	12.3	MS3运行No.选择	12
定位结束输出偏差门	-1.2	MS4运行No.选择	12
AREA1+方向位置[step]	-1234567	MS5运行No.选择	12
AREA1-方向位置[step]	-1234567		
AREA2+方向位置[step]	-1234567		
AREA2-方向位置[step]	-1234567		
AREA3+方向位置[step]	-1234567		
AREA3-方向位置[step]	-1234567		
MOVE输出最短时间[ms]	123		
LS接点设定	常开接点[N.O.]		
HOMES接点设定	常开接点[N.O.]		
SLIT接点设定	常开接点[N.O.]		
HOME-P输出功能选择	原点输出		

菜单

参数菜单2

I/O

电机

运行

原点返回

报警警告

返回

概要

显示、编辑 I/O 的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑 I/O 的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 更改 LS 接点设定、HOMES 接点设定、SLIT 接点设定时，请务必执行「Configuration」指令。  
不执行「Configuration」指令，将不能反映所更改的设定。  
「Configuration」指令在数据管理画面上可执行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

20/209

BCN-P5999-0417-2

5.3.6 参数 电机(B-31005)

1

电机

2014/09/10 09:44

5

6

号机号码 56 轴 AC

RUN电流[%]	123.4		
STOP电流[%]	12.3		
位置回路增益	12		
速度回路增益	123		
速度回路积分时间常数[ms]	123.4		
速度平滑调整[ms]	123		
移动平均时间[ms]	123		
平滑调整选择	速度平滑调整		
速度差增益1	123		
速度差增益2	123		
控制模式	正常模式		
平滑驱动	无效		

3

菜单 参数菜单2 I/O 电机 运行 原点返回 报警警告 返回

4

概要

显示、编辑电机的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑电机的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 更改滤波器、控制模式、平滑驱动时，请务必执行「Configuration」指令。  
不执行「Configuration」指令，将不能反映所更改的设定。  
「Configuration」指令在数据管理画面上可执行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.7 参数 运行(B-31006)

运行

2014/09/10 09:45

5

6

1

2

3

4

号机号码

56 轴 AC

通用加速[ms/kHz] or [s]	1234.567		
通用减速[ms/kHz] or [s]	1234.567		
起动速度[Hz]	1234567		
JOG运行速度[Hz]	1234567		
JOG加速[ms/kHz] or [s]	1234.567		
JOG起动速度[Hz]	1234567		
加速选择	通用		
加速单位	ms/kHz		
自动返回动作	无效		
自动返回运行速度[Hz]	1234567		
自动返回加速[ms/kHz] or [s]	1234.567		
自动返回起动速度[Hz]	1234567		
JOG移动量[step]	1234567		

菜单

参数菜单2

I/O

电机

运行

原点返回

报警警告

返回

3

4

概要

显示、编辑运行的相关参数。

详细

1.

显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2.

显示、编辑运行的相关参数。数值、文字为黄色时，代表其为初始值。

3.

切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4.

切换至上次显示画面。

5.

显示当前日期和时间。触摸即显示时钟设置窗口。

6.

显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 更改加速单位、自动恢复动作时，请务必执行「Configuration」指令。  
不执行「Configuration」指令，将不能反映所更改的设定。  
「Configuration」指令在数据管理画面上可执行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.8 参数 原点返回(B-31007)

1

原点返回

2014/09/10 09:46

5

6

号机号码 56 轴 AC

2

原点返回方法	2传感器方式		
原点返回运行速度[Hz]	1234567		
原点返回加减速[ms/kHz] or [s]	1234.567		
原点返回启动速度[Hz]	1234567		
原点返回偏置[step]	-1234567		
原点返回开始方向	-侧		
原点返回SLIT传感器检测	无效		
原点返回TIM信号检测	无效		
压推原点返回运行电流[%]	123.4		

3

菜单 参数菜单2 I/O 电机 运行 原点返回 报警警告 返回

4

概要

显示、编辑原点返回的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑原点返回的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.9 参数 报警&警告(B-31008)

1

报警&警告

2014/09/10 09:48

5

6

号机号码 56 轴 AC

过载报警[s]	12.3		
电源ON时位置偏差过大报警 [rev]	123.45		
原点返回未结束报警	无效		
电源OFF时位置偏差过大报警 [rev]	123.45		
过热警告[℃]	12		
过载警告[s]	12.3		
超速警告[r/min]	1234		
过压警告[V]	1234		
电压不足警告[V]	123		
电流ON时位置偏差过大警告 [rev]	123.45		

3

菜单 参数菜单2 I/O 电机 运行 原点返回 报警&警告 返回

4

概要

显示、编辑报警和警告的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑报警和警告的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

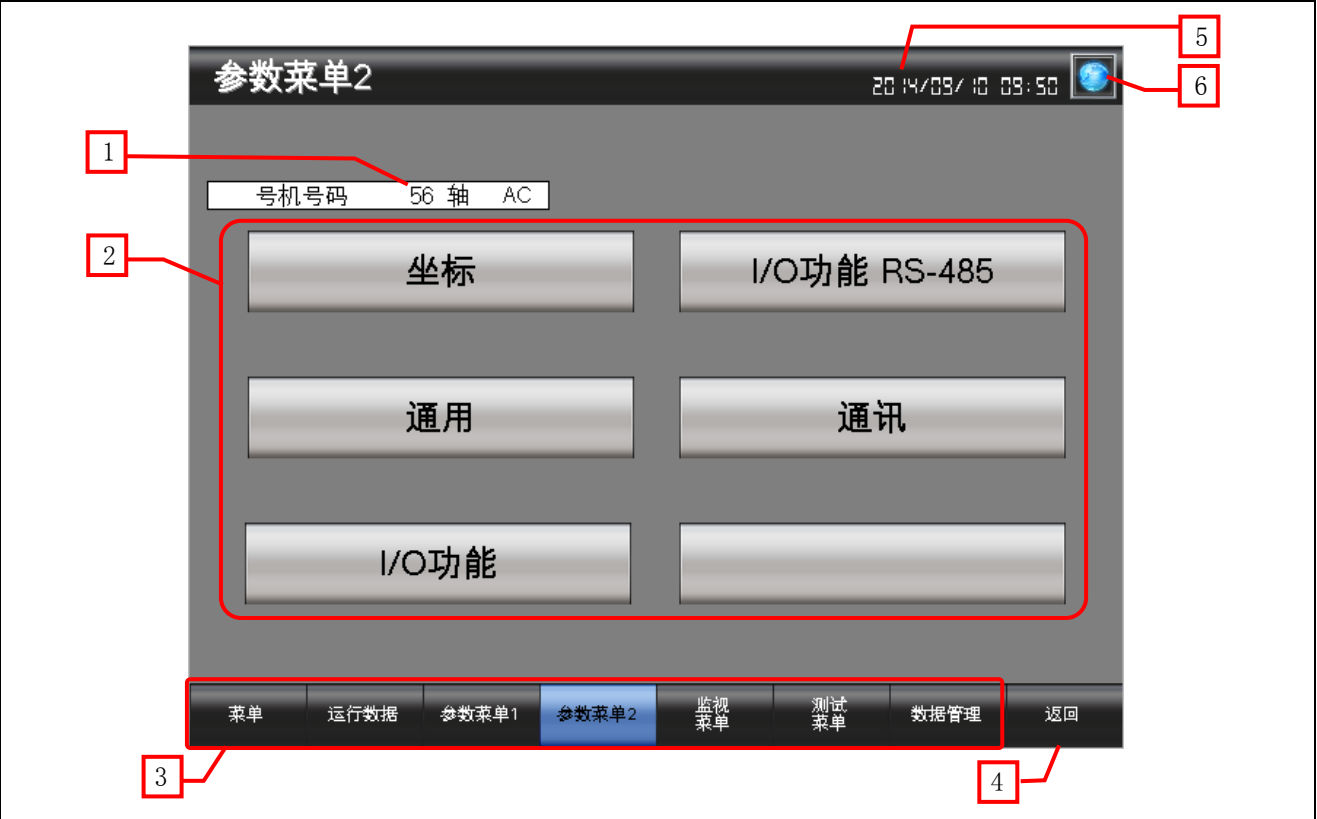
• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 更改原点恢复未结束的报警时，请务必执行「Configuration」指令。  
不执行「Configuration」指令，将不能反映所更改的设定。  
「Configuration」指令在数据管理画面上可执行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。



5.3.10 参数菜单 2(B-30004)



概要

参数菜单 2 画面。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 切换至各画面。
3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
4. 切换至上次显示画面。
5. 显示当前日期和时间。触摸即显示时钟设置窗口。
6. 显示语言设置窗口。

备注

- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.11 参数 坐标(B-31010)

1

坐标

2014/09/10 09:50

5

6

号机号码 56 轴 AC

电子减速机A	12345		
电子减速机B	12345		
电机旋转方向	+侧 = CCW		
软件超程	无效		
+软件极限[step]	-1234567		
-软件极限[step]	-1234567		
预置位置[step]	-1234567		
循环设定	无效		
循环设定范围[step]	1234567		

3

菜单 参数菜单1 坐标 通用 I/O功能 I/O功能 RS-485 通讯 返回

4

概要

显示、编辑坐标的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑坐标的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 更改电子减速机 A、电子减速机 B、电动机旋转方向、循环设定、循环设定范围时，请务必执行「Configuration」指令。不执行「Configuration」指令，将不能反映所更改的设定。  
「Configuration」指令在数据管理画面上可执行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.12 参数 通用(B-31011)

1

通用

2014/09/10 09:52

5

6

号机号码 56 轴 AC

数据设定器速度显示

数据设定器编辑

绝对位置备份系统

有符号

无效

无效

2

3

菜单

参数菜单1

坐标

通用

I/O功能

I/O功能 RS-485

通讯

返回

4

概要

显示、编辑通用的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑通用的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 更改绝对型备份系统时，请务必执行「Configuration」指令。  
不执行「Configuration」指令，将不能反映所更改的设定。  
「Configuration」指令在数据管理画面上可执行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.13 参数 I/O 功能(B-31012)

1

I/O功能

2014/09/10 09:52

5

6

号机号码

56 轴

AC

IN0输入功能选择	HOME	OUT0输出功能选择	HOME-P
IN1输入功能选择	START	OUT1输出功能选择	END
IN2输入功能选择	M0	OUT2输出功能选择	AREA1
IN3输入功能选择	M1	OUT3输出功能选择	READY
IN4输入功能选择	M2	OUT4输出功能选择	WNG
IN5输入功能选择	FREE	OUT5输出功能选择	ALM
IN6输入功能选择	STOP		
IN7输入功能选择	ALM-RST		
IN0输入接点设定	常开接点[N.O.]		
IN1输入接点设定	常开接点[N.O.]		
IN2输入接点设定	常开接点[N.O.]		
IN3输入接点设定	常开接点[N.O.]		
IN4输入接点设定	常开接点[N.O.]		
IN5输入接点设定	常开接点[N.O.]		
IN6输入接点设定	常开接点[N.O.]		
IN7输入接点设定	常开接点[N.O.]		

菜单

参数菜单1

坐标

通用

I/O功能

I/O功能 RS-485

通讯

返回

3

4

概要

显示、编辑 I/O 功能的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑 I/O 功能的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 更改 I/O 功能的参数时，请务必执行「Configuration」指令。  
不执行「Configuration」指令，将不能反映更改了信号的功能。  
「Configuration」指令，可在数据管理画面中执行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

28/209

BCN-P5999-0417-2

5.3.14 参数 I/O 功能 RS-485(B-31013)

1

I/O功能 RS-485

2014/09/10 10:01

5

6

号机号码

56 轴 AC

NET-IN0输入功能选择	M0	NET-OUT0输出功能选择	M0_R
NET-IN1输入功能选择	M1	NET-OUT1输出功能选择	M1_R
NET-IN2输入功能选择	M2	NET-OUT2输出功能选择	M2_R
NET-IN3输入功能选择	START	NET-OUT3输出功能选择	START_R
NET-IN4输入功能选择	HOME	NET-OUT4输出功能选择	HOME-P
NET-IN5输入功能选择	STOP	NET-OUT5输出功能选择	READY
NET-IN6输入功能选择	FREE	NET-OUT6输出功能选择	WNG
NET-IN7输入功能选择	未使用	NET-OUT7输出功能选择	ALM
NET-IN8输入功能选择	MS0	NET-OUT8输出功能选择	S-BSY
NET-IN9输入功能选择	MS1	NET-OUT9输出功能选择	AREA1
NET-IN10输入功能选择	MS2	NET-OUT10输出功能选择	AREA2
NET-IN11输入功能选择	SSTART	NET-OUT11输出功能选择	AREA3
NET-IN12输入功能选择	+JOG	NET-OUT12输出功能选择	TIM
NET-IN13输入功能选择	~JOG	NET-OUT13输出功能选择	MOVE
NET-IN14输入功能选择	FWD	NET-OUT14输出功能选择	END
NET-IN15输入功能选择	RVS	NET-OUT15输出功能选择	TLC

菜单

参数菜单1

坐标

通用

I/O功能

I/O功能 RS-485

通讯

返回

3

4

概要

显示、编辑 I/O 功能 RS-485 的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑 I/O 功能 RS-485 的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 更改 I/O 功能 RS-485 的参数时，请务必执行「Configuration」指令。  
不执行「Configuration」指令，将不能反映更改了信号的功能。  
「Configuration」指令，可在数据管理画面中执行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

29/209

BCN-P5999-0417-2

5.3.15 参数 通讯(B-31014)

1

通讯

2014/09/10 10:02

5

6

号机号码

56 轴 AC

通讯超时[ms]

12345

通讯异常报警[次]

12

2

3

菜单

参数菜单1

坐标

通用

I/O功能

I/O功能 RS-485

通讯

4

返回

概要

显示、编辑通讯的相关参数。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示、编辑通讯的相关参数。数值、文字为黄色时，代表其为初始值。

3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4. 切换至上次显示画面。

5. 显示当前日期和时间。触摸即显示时钟设置窗口。

6. 显示语言设置窗口。

备注

• 根据电源的类型，参数的设置范围会有所不同。详细内容请参照驱动器的使用说明书。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

30/209

BCN-P5999-0417-2

5.3.16 监视菜单(B-30005)



概要

监视菜单画面。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 切换至各画面。
3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
4. 切换至上次显示画面。
5. 显示当前日期和时间。触摸即显示时钟设置窗口。
6. 显示语言设置窗口。

备注

- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.17 监视 状态(B-31015)

1

状态

2014/09/10 10:04

8

9

号机号码

56 轴 AC

2

指令位置

-1234567890

[step]

指令速度

-1234

[r/min]

反馈位置

-1234567890

[step]

反馈速度

-1234

[r/min]

运行数据No.

-12

选择数据No.

12

3

当前的报警

00h:无报警

当前的警告

00h:无警告

4

复位绝对位置异常报警

报警复位

5

6

菜单

状态

I/O监视

I/O RS-485

报警记录

警告记录

手册显示

返回

7

概要

监视电机的状态。

详细

1.

显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2.

显示各项目的状态。

3.

显示当前发生中的报警和警告。

4.

复位绝对位置异常报警。复位后，执行原点返回运行等，请再次设置原点。

5.

复位当前发生中的报警。

6.

切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

7.

切换至上次显示画面。

8.

显示当前日期和时间。触摸即显示时钟设置窗口。

9.

显示语言设置窗口。

备注

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

32/209

BCN-P5999-0417-2



### 5.3.18 监视 I/O 监视(B-31016)



## 概要

监视 I/O。

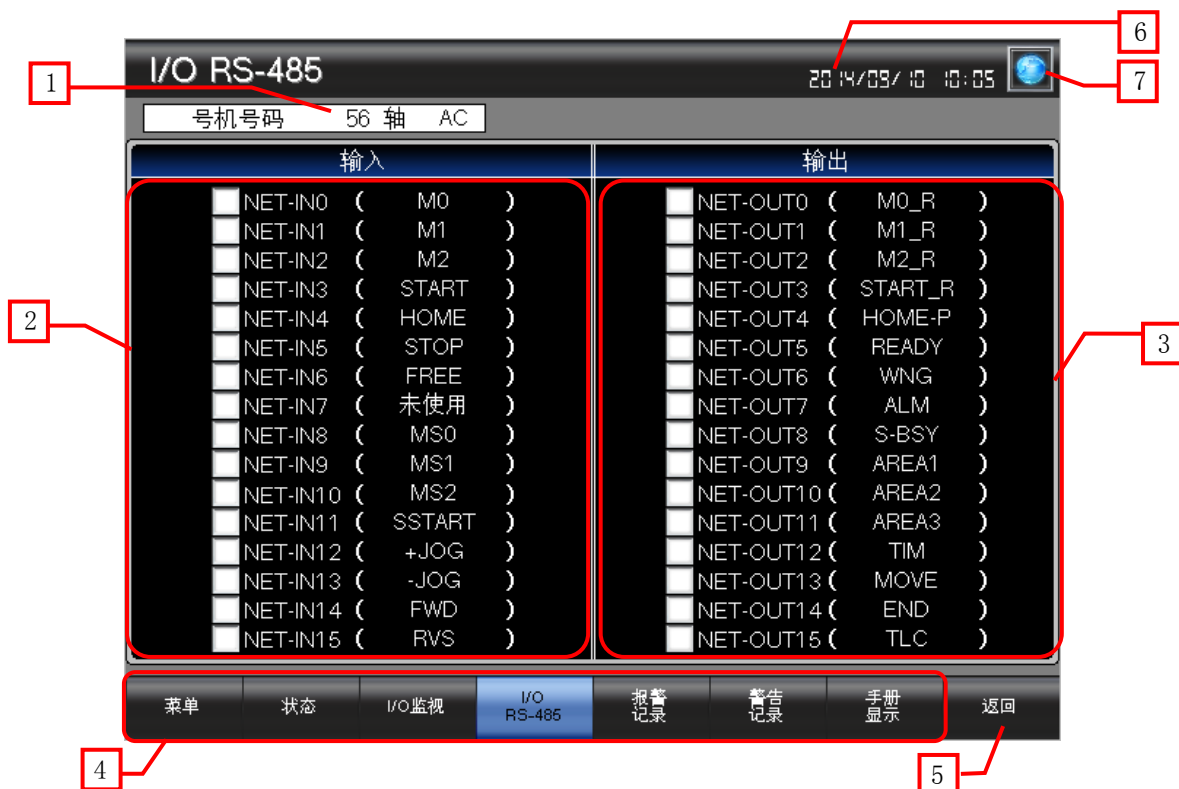
### 详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 显示 INPUT 的状态。
3. 显示 OUTPUT 的状态。
4. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
5. 切换至上次显示画面。
6. 显示当前日期和时间。触摸即显示时钟设置窗口。
7. 显示语言设置窗口。

## 备注

- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

### 5.3.19 监视 I/O RS-485 (B-31017)



## 概要

监视网络通信用的 I/O。

### 详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 显示 INPUT 的状态。
3. 显示 OUTPUT 的状态。
4. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
5. 切换至上次显示画面。
6. 显示当前日期和时间。触摸即显示时钟设置窗口。
7. 显示语言设置窗口。

## 备注

- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.20 监视 报警记录(B-31018)

报警记录

2014/09/10 10:06

9

10

号机号码 56 轴 AC

2

当前的报警 00h:无报警

当前的警告 00h:无警告

3

No.	报警信息
1	00h:无报警
2	00h:无报警
3	00h:无报警
4	00h:无报警
5	00h:无报警
6	00h:无报警
7	00h:无报警
8	00h:无报警
9	00h:无报警
10	00h:无报警

5

4

复位绝对位置异常报警

报警复位

清除报警记录

6

7

菜单

状态

I/O监视

I/O RS-485

报警记录

警告记录

手册显示

返回

8

概要

显示报警记录。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 显示当前发生中的报警和警告。

3. 显示报警记录。

4. 复位绝对位置异常报警。复位后，执行原点返回运行等，请再次设置原点。

5. 复位当前发生中的报警。

6. 清除报警记录。

7. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

8. 切换至上次显示画面。

9. 显示当前日期和时间。触摸即显示时钟设置窗口。

10. 显示语言设置窗口。

备注

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

35/209

BCN-P5999-0417-2

5.3.21 监视 警告记录(B-31019)

1

警告记录

2014/09/10 10:08

9

10

号机号码

56 轴 AC

2

当前的报警

00h:无报警

当前的警告

00h:无警告

3

No.	警告信息
1	00h:无警告
2	00h:无警告
3	00h:无警告
4	00h:无警告
5	00h:无警告
6	00h:无警告
7	00h:无警告
8	00h:无警告
9	00h:无警告
10	00h:无警告

4

复位绝对位置异常报警

报警复位

清除警告记录

5

6

7

菜单

状态

I/O监视

I/O RS-485

报警记录

警告记录

手册显示

返回

8

概要

显示警告记录。

详细

1.

显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2.

显示当前发生中的报警和警告。

3.

显示警告记录。

4.

复位绝对位置异常报警。复位后，执行原点返回运行等，请再次设置原点。

5.

复位当前发生中的报警。

6.

清除警告记录。

7.

切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

8.

切换至上次显示画面。

9.

显示当前日期和时间。触摸即显示时钟设置窗口。

10.

显示语言设置窗口。

备注

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。






5.3.22 手册显示(B-30500)



概要

显示与显示中的语言对应的手册。

详细

1. 手册显示是对应语言分别显示文件 ID 从 201 至 203 的文件。画面初次显示时，显示第 1 页。在触摸文件的状态下往 8 个方位拨动，文件即往其拨动方向滚动显示。拨动显示中的文件边端时，可以切换页码。通过双指张开/合拢操作，可依大/中/小 3 个阶段切换文件。
2. 操作显示中的文件。  
 : 放大/缩小显示中的文件。  
 : 左右滚动显示中的文件。  
 : 上下滚动显示中的文件。
3. 操作显示中的文件页。  
 : 显示正在显示中的文件页。触摸数值后可以更改页码。  
 : 对显示中的文件进行页发送/页返回。
4. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
5. 切换至上次显示画面。
6. 显示当前日期和时间。触摸即显示时钟设置窗口。
7. 显示语言设置窗口。

## 备注

- 手册显示的文件遵从显示语言切换。注释组号和语言、文件 ID 对应，如下表所示。

注释组号	语言	文件 ID 列号
1	中文(简体)	201
2	日语	202
3	英语	203

- GOT 启动时，通过工程脚本将文件页码设置为「1」以及将文件 ID 设置为「201」。关于脚本的详细内容，请参照「5.6 脚本一览表」。
- 页码传送开关通过对象脚本不超过总页数。关于脚本的详细内容，请参照「5.6 脚本一览表」。
- 手册显示用的文件数据由用户制作。有关详细请参照「6. 关于手册显示」。
- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5. 3. 23 测试菜单(B-30006)



概要

测试菜单画面。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 切换至各画面。
3. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
4. 未使用的画面切换开关。
5. 显示当前日期和时间。触摸即显示时钟设置窗口。
6. 显示语言设置窗口。

备注

- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.24 测试 I/O 测试(B-31020)



概要

输入的确认为输出的测试。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。
2. 触摸勾选框即开始 I/O 测试。
3. 显示 INPUT 的状态。
4. 显示 OUTPUT 的状态。触摸开关可确认 OUT0~OUT5 输出的状态。
5. 复位当前发生中的报警。
6. 显示当前发生中的报警
7. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
8. 未使用的画面切换开关。
9. 显示当前日期和时间。触摸即显示时钟设置窗口。
10. 显示语言设置窗口。

备注

- I/O 测试开始或结束时，电机的励磁会有一瞬间被切断，因此在用作升降轴等的情况下需要额外注意。而且，电机在瞬间切断励磁后再次励磁，位置会发生偏移。如果要追求位置精度，建议在 I/O 测试结束后立刻执行原点返回运行。
- I/O 测试中，不能切换到其他画面，也不能更改号机号码。
- 通过画面脚本执行 I/O 测试。关于脚本的详细内容，请参照「5.6 脚本一览表」。
- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。



5.3.25 测试 运行(B-31021)

1

测试运行

2014/09/05 16:22

5

号机号码

56 轴 AC

6

2

开始测试运行

指令位置

-1234567890

[step]

指令速度

-1234

[r/min]

运行数据No.

12

位置

-1234567

[step]

运行方式

INC

运行速度

1234567

[Hz]

定位运行

原点返回运行

位置预置

停止

JOG操作

微动操作

最小移动量

1234567

[step]

3

菜单

I/O测试

测试运行

多点运行

4

概要

电机的测试运行。

详细

1.

显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2.

勾选开始测试运行即可开始测试运行。

指令位置

:

显示指令位置。

指令速度

:

显示指令速度。

运行数据 No.

:

选择运行数据 No.。

位置

:

显示位置。

运行方式

:

显示运行方式。

运行速度

:

显示运行速度。

原点返回运行

:

开始原点返回运行。

定位运行

:

按照运行数据 No. 中所选的运行数据执行定位运行。

位置预置

:

将预置值设置到指令位置中。可以在「参数 坐标」的项目「预置位置」中更改预置值。

停止

:

停止运行中的电机。

:

持续按压开关期间，连续运行正转/反转。

通过运行数据 No. 选择的 No. 便是运行速度、加速、减速。

:

调整电机的位置。电机仅移动最小移动量中所设置的移动量。

最小移动量

:

可设置电机动作的最小移动量。

3.

切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

4.

未使用的画面切换开关。

5.

显示当前日期和时间。触摸即显示时钟设置窗口。

6.

显示语言设置窗口。

41/209

BCN-P5999-0417-2

#### 备注

- 测试运行开始或结束时，电机的励磁会有一瞬间被切断，因此在用作升降轴等的情况下需要额外注意。而且，电机在瞬间切断励磁后再次励磁，位置会发生偏移。如果要追求位置精度，建议在测试运行结束后立刻执行原点返回运行。
- 在测试运行中，加减速选择参数的设定为独立时，将更改为加速、减速选择的运行数据设定。测试运行中，不能切换到其他画面，也不能更改号机号码。
- 通过画面脚本执行测试运行。关于脚本的详细内容，请参照「5.6 脚本一览表」。
- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。



5. 3. 26 测试 多点运行(B-31022~31029)



概要

多个轴的电机的测试运行。

详细

- 1. 选择执行电机的测试运行的轴。
- 2. 触摸各个轴开关，所选轴即可测试运行。
- 3. 显示发生中的报警。
- 4. 执行各个轴的测试运行。
  - 数据 No. : 选择运行数据 No. 。
  -  : 持续按压开关期间，连续运行正转/反转。
  -  : 通过运行数据 No. 选择的 No. 便是运行速度、加速、减速。
  - 定位运行 : 调整电机的位置。电机仅移动最小移动量中所设置的移动量。
  - 位置预置 : 按照运行数据 No. 中所选的运行数据执行定位运行。
  - 停止 : 将预置值设置到指令位置中。可以在「参数 坐标」的项目「预置位置」中更改预置值。
  - 原点返回运行 : 停止运行中的电机。
  - 报警复位 : 开始原点返回运行。
- 5. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。
- 6. 未使用的画面切换开关。
- 7. 显示当前日期和时间。触摸即显示时钟设置窗口。
- 8. 显示语言设置窗口。

#### 备注

- 测试运行开始或结束时，电机的励磁会有一瞬间被切断，因此在用作升降轴等的情况下需要额外注意。而且，电机在瞬间切断励磁后再次励磁，位置会发生偏移。如果要追求位置精度，建议在测试运行结束后立刻执行原点返回运行。
- 在测试运行中，加减速选择参数的设定为独立时，将更改为加速、减速选择的运行数据设定。
- 测试运行中，不能切换到其他画面，也不能更改号机号码。
- 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

5.3.27 系统 数据管理(B-31030)

1

数据管理

2014/09/10 10:20

9

10

号机号码 1 轴 DC

2

参数初始化

运行数据/参数的全数据初始化

全数据初始化

3

进行参数的重新计算和设定

Configuration

NV存储器

保存运行数据和参数到驱动器(保存到不挥发存储器)

批量写入 RAM→NV存储器

4

读取保存在驱动器的NV存储器(不挥发存储器)中的运行数据和参数到RAM

批量读出 NV存储器→RAM

5

导入/导出

更改后的运行数据和参数可以保存到USB存储器或SD卡。(也可以从USB存储器、SD卡中读取)

配方文件一览表

6

※长按各设定按钮2秒以上进行动作。

7

菜单

运行数据

参数菜单1

参数菜单2

监视菜单

测试菜单

数据管理

8

返回

概要

驱动器的初始化、Configuration 以及运行数据及参数的保存、读取。

详细

1. 显示所监视驱动器的号机号码。触摸即可显示轴切换窗口。

2. 初始化保存在驱动器的 NV 存储器中的全部运行数据及参数。

3. 执行软件复位。某些参数，不执行软件复位就不能反映。

4. 将保存在 RAM 中的运行数据及参数保存至 NV 存储器。

5. 将保存在 NV 存储器中的运行数据及参数读取至 RAM。

6. 显示配方文件一览表窗口。运行数据及参数保存至 USB 存储器或 SD 卡；或者从 USB 存储器或 SD 卡读取运行数据及参数。

7. 切换至各画面。蓝色开关为当前显示的画面，显示中的画面不被切换。

8. 切换至上次显示画面。

9. 显示当前日期和时间。触摸即显示时钟设置窗口。

10. 显示语言设置窗口。

备注

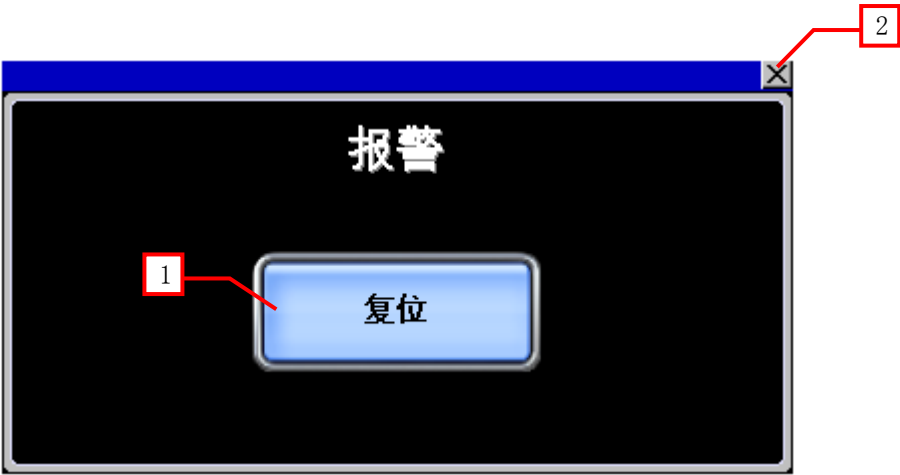
• 长按各设定开关 2 秒以上进行动作。

• 执行「Configuration」指令后，电机的励磁会有一瞬间被切断，因此在用作升降轴等的情况下需要额外注意。而且，电机在瞬间切断励磁后再次励磁，位置会发生偏移。如果要追求位置精度，建议在操作「Configuration」指令结束后立刻执行原点返回运行。

• 系统报警发生时，在画面下方将显示报警信息。触摸信息的左端时，显示位置依照画面上方、画面中央、画面下方的顺序切换。触摸其它地方时，显示报警复位窗口。

• 关于运行数据及参数的保存、读取的详细内容，请参照「8. 关于对 USB 存储器、SD 卡存取运行数据及参数」。

5.3.28 报警复位(W-30001)



概要

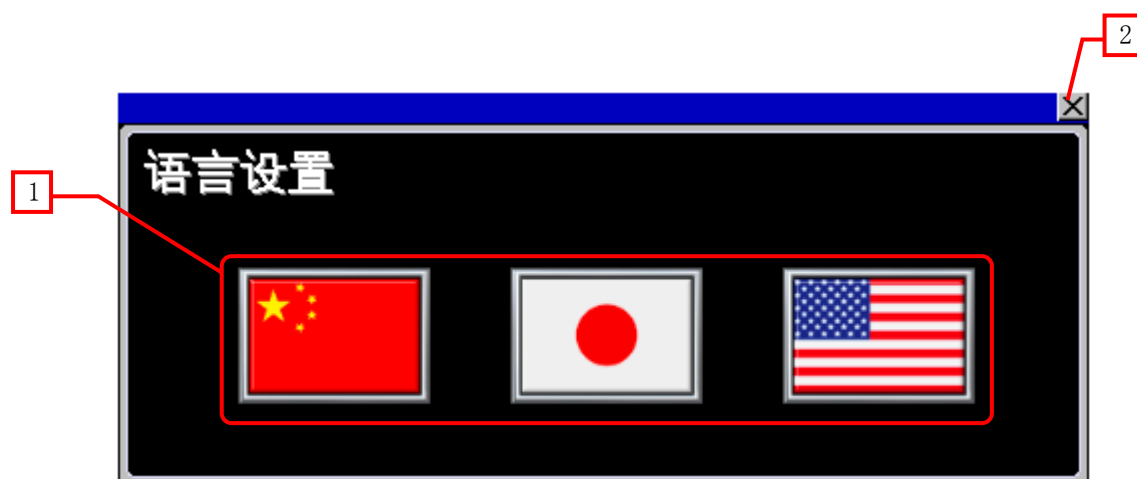
复位系统报警。

详细

1. 复位系统报警，并在 1 秒后关闭窗口画面。
2. 关闭窗口画面。

备注

### 5.3.29 语言设置(W-30002)



#### 概要

选择 GOT 的显示语言。

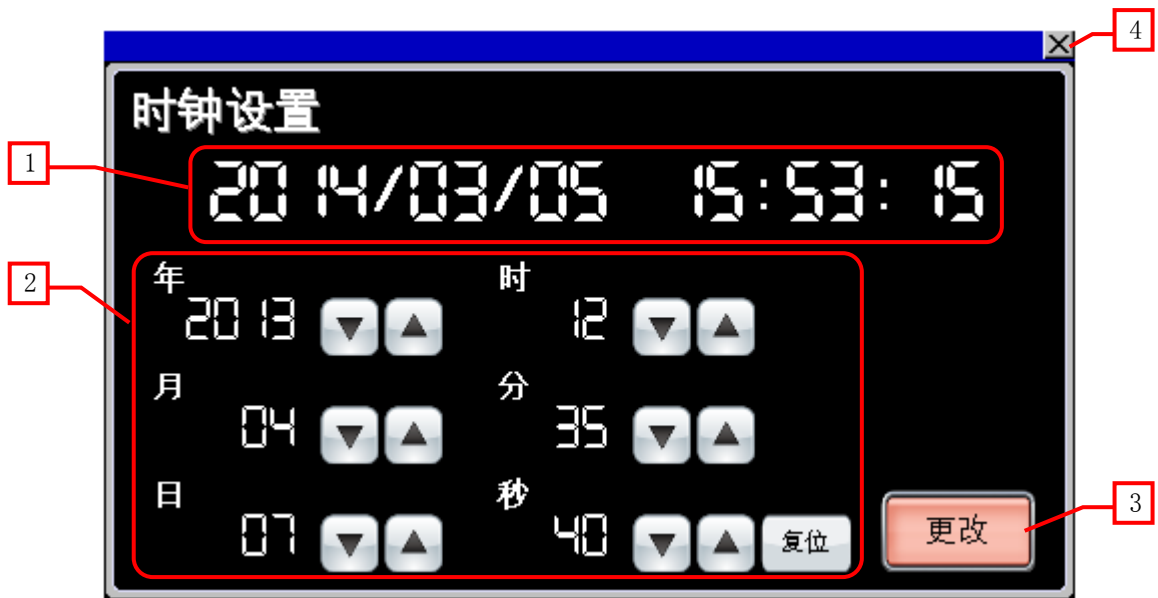
#### 详细

1. 切换语言，并关闭窗口画面。
2. 关闭窗口画面。

#### 备注

- 相应显示语言，系统语言与手册显示的文件 ID 同步切换设置。

5.3.30 时钟设置(W-30003)



概要

更改 GOT 的时钟数据。

详细

1. 显示当前日期和时间。
2. 通过 开关设置想更改的日期和时间。长按 开关将连续进行增减。复位开关复位秒。
3. 将设置的日期和时间反映到 GOT 的时钟数据中，并在 1 秒后关闭窗口画面。
4. 关闭窗口画面。

备注

- 日期和时间的初始值为窗口画面显示时的日期和时间。
- 更改日期和时间的年/月/日/时/分/秒的数值显示中设置了对象脚本。  
详细内容，请参照「5.6 脚本一览表」。



5. 3. 31 轴切换 (W-30004)



概要

切换号机选择和电源类型。

详细

1. 选择监视的号机。
2. 指定监视的号机的驱动器的电源类型。
3. 不反映设置，关闭窗口画面。
4. 反映设置，关闭窗口画面。没有选择号机选择、电源类型时，开关不能动作。

备注

- 通过画面脚本设置号机选择、电源类型。关于脚本的详细内容，请参照「5.6 脚本一览表」。

5. 3. 32 输入运行数据(W-32001)

No. 12	
运行方式	<input type="button" value="增量"/> <input type="button" value="绝对位置"/>
位置	-1234567 [step]
运行速度	1234567 [Hz]
运行功能	<input type="button" value="单独"/> <input type="button" value="连结"/> <input type="button" value="连结2"/> <input type="button" value="压推"/>
压推电流	12.3 [%]
停留时间	12.345 [s]
顺序定位	<input type="button" value="无效"/> <input type="button" value="有效"/>
加速	1234.567 [ms/kHz]
减速	1234.567 [ms/kHz]

最大值		8388607	
最小值		-8388608	
DEL	AC	<<	>>
7	8	9	+/-
4	5	6	
1	2	3	Enter
0		.	

取消 设定

概要

显示、编辑运行数据。

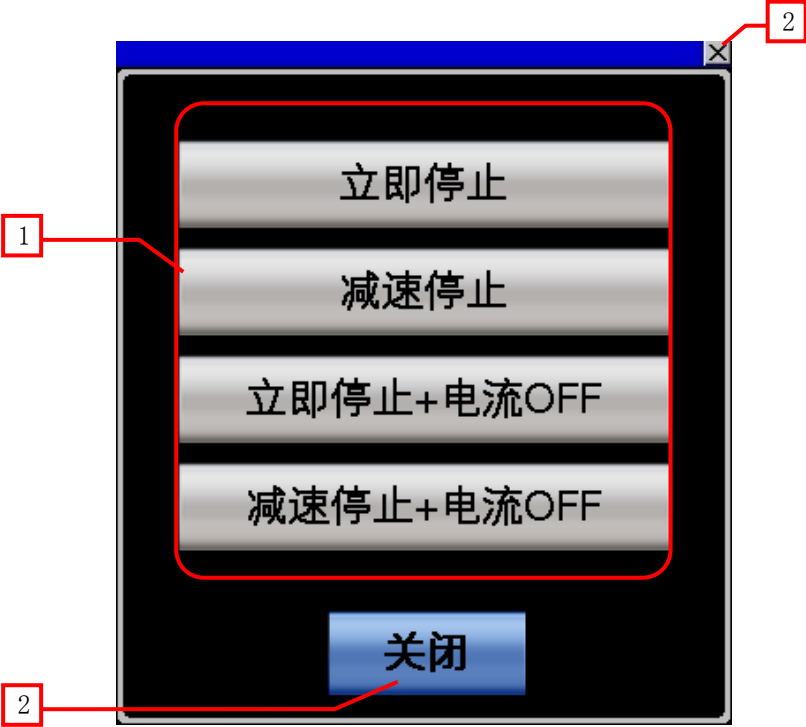
详细

1. 显示、编辑运行数据。
2. 输入用数字键盘。
3. 不反映编辑过的数据，关闭窗口画面。
4. 反映编辑过的数据，关闭窗口画面。

备注

- 通过画面脚本执行运行数据的读取，写入。关于脚本的详细内容，请参照「5.6 脚本一览表」。

5. 3. 33 STOP 输入停止方法(W-32002)



概要

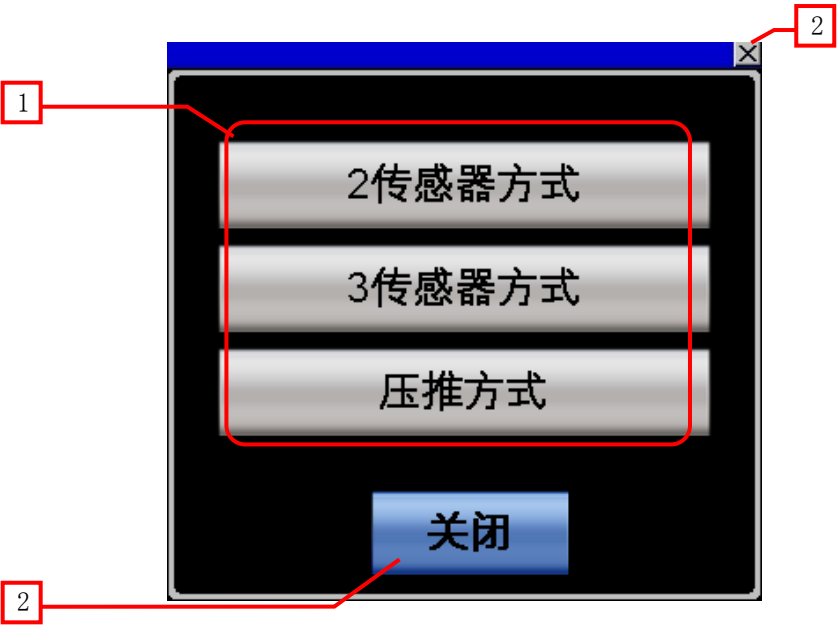
显示、设置 STOP 输入停止方法的内容。

详细

1. 显示、设置 STOP 输入停止方法的内容。
2. 关闭窗口画面。

备注

5. 3. 34 原点返回方法(W-32003)



概要

显示、设置原点返回方法的内容。

详细

1. 显示、设置原点返回方法的内容。
2. 关闭窗口画面。

备注

5. 3. 35 IN 输入功能选择 (W-32004)

未使用	STOP	R11
FWD	ALM-RST	R12
RVS	P-PRESET	R13
HOME	P-CLR	R14
START	HMI	R15
SSTART	R0	M0
+JOG	R1	M1
-JOG	R2	M2
MS0	R3	M3
MS1	R4	M4
MS2	R5	M5
MS3	R6	
MS4	R7	
MS5	R8	
FREE	R9	
C-ON	R10	

取消 设定

**概要**

显示、设置 IN 输入功能选择的内容。

**详细**

1. 显示、设置 IN 输入功能选择的内容。
2. 不反映设置，关闭窗口画面。
3. 反映设置，关闭窗口画面。

**备注**

- 通过画面脚本执行设置内容的写入。关于脚本的详细内容，请参照「5.6 脚本一览表」。

5. 3. 36 NET-IN 输入功能选择 (W-32005)



概要

显示、设置 NET-IN 输入功能选择的内容。

详细

1. 显示、设置 NET-IN 输入功能选择的内容。
2. 不反映设置，关闭窗口画面。
3. 反映设置，关闭窗口画面。

备注

- 通过画面脚本执行设置内容的写入。关于脚本的详细内容，请参照「5.6 脚本一览表」。

5. 3. 37 OUT/NET-OUT 输出功能选择(W-32006)



概要

显示、设置 OUT/NET-OUT 输出功能选择的内容。

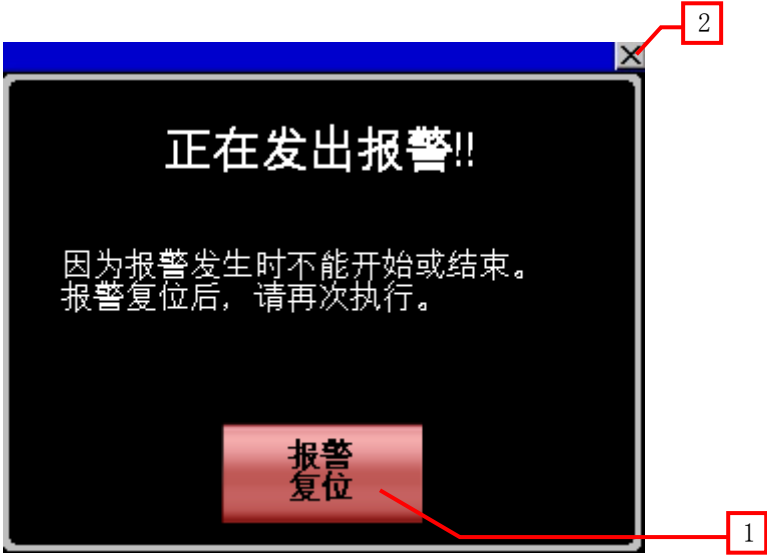
详细

1. 显示、设置 OUT/NET-OUT 输出功能选择的内容。
2. 不反映设置，关闭窗口画面。
3. 反映设置，关闭窗口画面。

备注

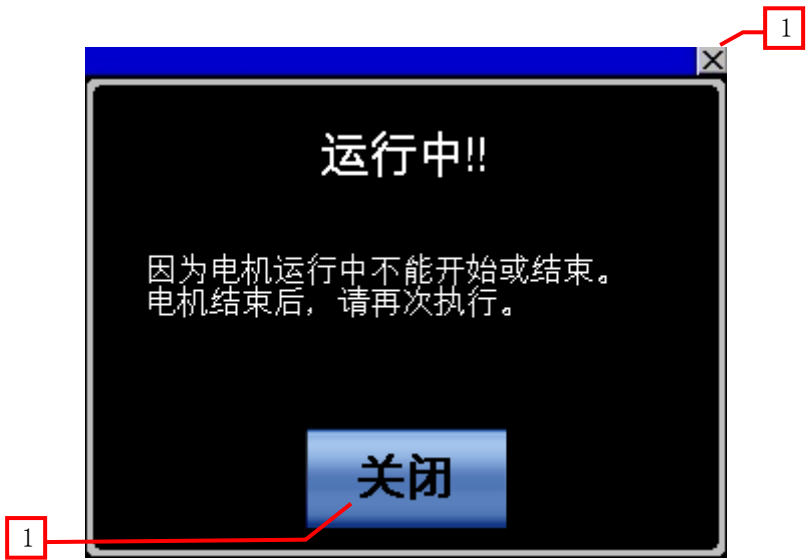
- 关于 MPS，仅在电源类型为 AC 时可以设置。
- 通过画面脚本执行设置内容的写入。关于脚本的详细内容，请参照「5.6 脚本一览表」。

5. 3. 38 报警发生中确认(W-32007)

<div data-bbox="509 309 1279 862"></div>	
<div data-bbox="162 1019 1489 1059"><b>概要</b></div>	
<div data-bbox="162 1059 1489 1160"><p>有报警发生的时候，执行或结束示教运行、各测试运行时会弹出报警发生中窗口。</p></div>	
<div data-bbox="162 1160 1489 1200"><b>详细</b></div>	
<div data-bbox="162 1200 1489 1749"><ol style="list-style-type: none"><li>1. 复位发生中的报警，关闭窗口画面。</li><li>2. 关闭窗口画面。</li></ol></div>	
<div data-bbox="162 1749 1489 1789"><b>备注</b></div>	
<div data-bbox="162 1789 1489 2065"></div>	



5. 3. 39 电机运行中确认(W-32008)



概要

有电机运行的时候，执行或结束示教运行、各测试运行会弹出电机运行中窗口。

详细

1. 关闭窗口画面。

备注

## 5.4 使用软元件一览表

画面上的开关和指示灯等中进行设置的一部分软元件，在[公共设置]中也可能被设置。批量更改此类软元件时，推荐使用[批量更改]。关于[批量更改]的详细内容，请参照「GT Designer3 (GOT2000) 帮助」。

### 5.4.1 连接机器的软元件

类型	软元件编号	用途
位	400126. b0	驱动器输入指令[M0]
	400126. b1	驱动器输入指令[M1]
	400126. b2	驱动器输入指令[M2]
	400126. b3	驱动器输入指令[START]
	400126. b4	驱动器输入指令[HOME]
	400126. b5	驱动器输入指令[STOP]
	400126. b6	驱动器输入指令[FREE]
	400126. b7	驱动器输入指令[未使用]
	400126. b8	驱动器输入指令[MS0]
	400126. b9	驱动器输入指令[MS1]
	400126. b10	驱动器输入指令[MS2]
	400126. b11	驱动器输入指令[SSTART]
	400126. b12	驱动器输入指令[+JOG]
	400126. b13	驱动器输入指令[-JOG]
	400126. b14	驱动器输入指令[FWD]
	400126. b15	驱动器输入指令[RVS]
	400128. b0	驱动器输出指令[M0_R]
	400128. b1	驱动器输出指令[M1_R]
	400128. b2	驱动器输出指令[M2_R]
	400128. b3	驱动器输出指令[START_R]
	400128. b4	驱动器输出指令[HOME-P]
	400128. b5	驱动器输出指令[READY]
	400128. b6	驱动器输出指令[WNG]
	400128. b7	驱动器输出指令[ALM]
	400128. b8	驱动器输出指令[S-BSY]
	400128. b9	驱动器输出指令[AREA1]
	400128. b10	驱动器输出指令[AREA2]
	400128. b11	驱动器输出指令[AREA3]
	400128. b12	驱动器输出指令[TIM]
	400128. b13	驱动器输出指令[MOVE]
	400128. b14	驱动器输出指令[END]
	400128. b15	驱动器输出指令[TLC]
	400213. b0	直接 I/O、电磁制动的状态[OUT0]
	400213. b1	直接 I/O、电磁制动的状态[OUT1]
	400213. b2	直接 I/O、电磁制动的状态[OUT2]
	400213. b3	直接 I/O、电磁制动的状态[OUT3]
	400213. b4	直接 I/O、电磁制动的状态[OUT4]
	400213. b5	直接 I/O、电磁制动的状态[OUT5]
	400214. b0	直接 I/O、电磁制动的状态[+LS]
	400214. b1	直接 I/O、电磁制动的状态[-LS]
	400214. b2	直接 I/O、电磁制动的状态[HOMES]
	400214. b3	直接 I/O、电磁制动的状态[SLIT]
	400214. b6	直接 I/O、电磁制动的状态[IN0]
	400214. b7	直接 I/O、电磁制动的状态[IN1]
	400214. b8	直接 I/O、电磁制动的状态[IN2]
	400214. b9	直接 I/O、电磁制动的状态[IN3]

类型	软元件编号	用途
位	400214. b10	直接 I/O、电磁制动的状态 [IN4]
	400214. b11	直接 I/O、电磁制动的状态 [IN5]
	400214. b12	直接 I/O、电磁制动的状态 [IN6]
	400214. b13	直接 I/O、电磁制动的状态 [IN7]
	400386. b0	Alarm 的复位
	400388. b0	绝对位置异常 Alarm 复位
	400390. b0	Alarm 履历的清除
	400392. b0	Warning 履历的清除
	400396. b0	执行 P-PRESET
	400398. b0	执行 Configuration
	400400. b0	全部数据初始化
	400402. b0	NV 存储器全部读取
	400404. b0	NV 存储器全部写入
	400516. b0	硬件超程
	400518. b0	超程动作
	400538. b0	LS 接点设定
	400540. b0	HOMES 接点设定
	400542. b0	SLIT 接点设定
	400654. b0	加减速选择
	400656. b0	加减速单位
	400716. b0	原点返回开始方向
	400718. b0	原点返回 SLIT 传感器检测
	400720. b0	原点返回 TIM 信号检测
	400778. b0	原点返回未结束 Alarm
	400902. b0	电动机旋转方向
	400904. b0	软件超程
	400912. b0	循环设定
	400962. b0	数据设定器速度显示
	400964. b0	数据设定器编辑
	400966. b0	绝对位置备份系统
	404110. b0	HOME-P 输出功能选择
	404130. b0	平滑调整选择
	404136. b0	控制模式
	404138. b0	平滑驱动
	404162. b0	自动返回动作
	404386. b0	IN0 输入接点设定
	404388. b0	IN1 输入接点设定
	404390. b0	IN2 输入接点设定
	404392. b0	IN3 输入接点设定
	404394. b0	IN4 输入接点设定
	404396. b0	IN5 输入接点设定
	404398. b0	IN6 输入接点设定
	404400. b0	IN7 输入接点设定
字	400126	驱动器输入指令 (下位)
	400129	当前的 Alarm (上位)
	400130	当前的 Alarm (下位)
	400132+2n (n=0~9)	Alarm 履历 1~10 (上位)
	400152	当前的 Warning (下位)
	400154+2n (n=0~9)	Warning 履历 1~10 (上位)
	400195	当前的选择数据 No. (上位)
	400196	当前的选择数据 No. (下位)
	400197	当前的运行数据 No. (上位)
	400199	指令位置 (上位)

类型	软元件编号	用途
字	400201	指令速度(上位)
	400205	反馈位置(上位)
	400207	反馈速度(上位)
	400395	执行 P-PRESET(上位)
	400397	Configuration(上位)
	400513	STOP 输入停止方法(上位)
	400514	STOP 输入停止方法(下位)
	400515	硬件超程(上位)
	400517	超程动作(上位)
	400519	定位结束输出宽度(上位)
	400521	定位结束输出偏置(上位)
	400523	AREA1 +方向位置(上位)
	400525	AREA1 -方向位置(上位)
	400527	AREA2 +方向位置(上位)
	400529	AREA2 -方向位置(上位)
	400531	AREA3 +方向位置(上位)
	400533	AREA3 -方向位置(上位)
	400535	MOVE 输出最短时间(上位)
	400537	LS 接点设定(上位)
	400539	HOMES 接点设定(上位)
	400541	SLIT 接点设定(上位)
	400577	RUN 电流(上位)
	400579	STOP 电流(上位)
	400581	位置回路增益(上位)
	400583	速度回路增益(上位)
	400585	速度回路积分时间常数 (上位)
	400587	速度平滑调整(上位)
	400589	移动平均时间(上位)
	400641	通用加速(上位)
	400643	通用减速(上位)
	400645	起动速度(上位)
	400647	JOG 运行速度(上位)
	400649	JOG 加减速(上位)
	400651	JOG 起动速度(上位)
	400653	加减速选择(上位)
	400655	加减速单位(上位)
	400656	加减速单位(下位)
	400705	原点返回方法(上位)
	400706	原点返回方法(下位)
	400707	原点返回运行速度(上位)
	400709	原点返回加速度(上位)
	400711	原点返回起动速度(上位)
	400713	原点返回偏置(上位)
	400715	原点返回开始方向(上位)
	400717	原点返回 SLIT 传感器检测(上位)
	400719	原点返回 TIM 信号检测(上位)
	400721	压推原点返回运行电流(上位)
	400769	过载 Alarm(上位)
	400771	电流 ON 时位置偏差过大 Alarm(上位)
	400777	原点返回未结束 Alarm(上位)
	400833	过热 Warning(上位)
	400835	过载 Warning(上位)
	400837	超速 Warning(上位)

类型	软元件编号	用途
字	400839	过压 Warning(上位)
	400841	电压不足 Warning(上位)
	400843	电流 ON 时位置偏差过大 Warning(上位)
	400897	电子减速机 A(上位)
	400899	电子减速机 B(上位)
	400901	电动机旋转方向(上位)
	400903	软件超程(上位)
	400905	+软件极限(上位)
	400907	-软件极限(上位)
	400909	预置位置(上位)
	400911	循环设定(上位)
	400913	循环设定范围(上位)
	400961	数据设定器速度显示(上位)
	400963	数据设定器显示(上位)
	400965	绝对位置备份系统(上位)
	401025+2n (n=0~63)	位置 No. 0~63(上位)
	401153+2n (n=0~63)	运行速度 No. 0~63(上位)
	401281+2n (n=0~63)	运行方式 No. 0~63 (上位)
	401282+2n (n=0~63)	运行方式 No. 0~63(下位)
	401409+2n (n=0~63)	运行功能 No. 0~63(上位)
	401410+2n (n=0~63)	运行功能 No. 0~63(下位)
	401537+2n (n=0~63)	加速 No. 0~63(上位)
	401665+2n (n=0~63)	减速 No. 0~63(上位)
	401793+2n (n=0~63)	压推电流 No. 0~63(上位)
	401921+2n (n=0~63)	顺序定位 No. 0~63(上位)
	401922+2n (n=0~63)	顺序定位 No. 0~63(下位)
	402049+2n (n=0~63)	停留时间 No. 0~63(上位)
	404097+2n (n=0~5)	MS0 运行 No. 选择(上位)~MS5 运行 No. 选择(上位)
	404109	HOME-P 输出功能选择(上位)
	404129	平滑调整选择(上位)
	404131	速度差增益 1(上位)
	404133	速度差增益 2(上位)
	404135	控制模式(上位)
	404137	平滑驱动(上位)
	404161	自动返回动作(上位)
	404163	自动返回运行速度(上位)
	404165	自动返回加减速(上位)
	404167	自动返回起动速度(上位)
	404169	JOG 运行(上位)
	404225	电流 OFF 时位置偏差过大 Alarm(上位)
	404353+2n (n=0~7)	IN0 输入功能选择(上位)~IN7 输入功能选择(上位)
	404354+2n (n=0~7)	IN0 输入功能选择(下位)~IN7 输入功能选择(下位)
	404385+2n (n=0~7)	IN0 输入接点设定(上位)~IN7 输入接点设定(上位)
	404417+2n (n=0~5)	OUT0 输出功能选择(上位)~OUT5 输出功能选择(上位)
	404418+2n (n=0~5)	OUT0 输出功能选择(下位)~OUT5 输出功能选择(下位)
	404449+2n (n=0~15)	NET-IN0 输入功能选择(上位)~NET-IN15 输入功能选择(上位)
	404450+2n (n=0~15)	NET-IN0 输入功能选择(下位)~NET-IN15 输入功能选择(下位)
	404481+2n (n=0~15)	NET-OUT0 输出功能选择(上位)~NET-OUT15 输出功能选择(上位)
	404482+2n (n=0~15)	NET-OUT0 输出功能选择(下位)~NET-OUT15 输出功能选择(下位)
	404609	通信超时(上位)
	404611	通信异常 Alarm(下位)

#### 5.4.2 GOT 内部软元件

类型	软元件编号	用途
位	GB40	脚本触发
	GB61000	状态标志
	GB61001	号机选择状态标志
	GB61002	电源类型状态标志
	GB61010	电源类型判别标志
	GB62000	显示运行数据上滚动
	GB62001	显示运行数据下滚动
	GB62002	运行数据输入的读取触发
	GB62003	运行数据输入的写入触发
	GB62004	I/O 功能选择的写入触发
	GB62005	OUT/NET-OUT 判别位
	GB62006	重叠窗口显示中标志
	GB62007	画面转换禁止联锁用标志
	GB62008	多点运行中联锁用标志
	GB62009	多点起动中/结束中标志
	GB62010	开始/结束前确认触发
	GB62011	报警发生状态确认标志
	GB62012	运行中确认标志
	GB62013	运行准备开始触发
	GB62014	运行准备中标志
	GB62015	触摸面板运行中标志
	GB62016	运行结束开始触发
	GB62017	运行结束标志
	GB62018	I/O 测试的标志
	GB62019	位置确定触发
	GB62020+10n (n=0~30)	开始/结束前确认触发(1 轴~31 轴)
	GB62021+10n (n=0~30)	报警发生状态确认标志(1 轴~31 轴)
	GB62022+10n (n=0~30)	运行中确认标志(1 轴~31 轴)
	GB62023+10n (n=0~30)	运行准备开始触发(1 轴~31 轴)
	GB62024+10n (n=0~30)	运行准备中标志(1 轴~31 轴)
	GB62025+10n (n=0~30)	触摸面板运行中标志(1 轴~31 轴)
	GB62026+10n (n=0~30)	运行结束开始触发(1 轴~31 轴)
	GB62027+10n (n=0~30)	运行结束标志(1 轴~31 轴)
	GD60031. b0	报警复位
	GD61020. b0~b5	运行数据输入画面的运行号码的 No. 显示
	GD61200+100n. b0 (n=0~30)	运行号码的 No. 显示(1 轴~31 轴)
	GD61200+100n. b1 (n=0~30)	运行号码的 No. 显示(1 轴~31 轴)
	GD61200+100n. b2 (n=0~30)	运行号码的 No. 显示(1 轴~31 轴)
	GD61200+100n. b3 (n=0~30)	运行号码的 No. 显示(1 轴~31 轴)
	GD61200+100n. b4 (n=0~30)	运行号码的 No. 显示(1 轴~31 轴)
	GD61200+100n. b5 (n=0~30)	运行号码的 No. 显示(1 轴~31 轴)
	GS281. b0~GS281. b15	异常站信息(CH1)
	GS512. b0	时间更改信息
	GS531. b0~GS532. b15	监视站断开(CH1)
字	GD60000	基本画面切换
	GD60001	重叠窗口 1 画面切换
	GD60004	重叠窗口 2 画面切换
	GD60021	语言切换
	GD60022	系统语言切换
	GD60031、GD60041	系统信息
	GD60042	当前光标显示用户 ID

类型	软元件编号	用途
字	GD60080～GD60082	文件显示
	GD61000	号机号码输入
	GD61001～GD61008	运行数据 No. 的号码显示(0 行～7 行)
	GD61010	上下滚动计数器值
	GD61011	Modbus 地址的偏置值
	GD61012	运行数据 No. 的触摸位置信息
	GD61015	示教运行时的指令位置、最小移动量显示的 Modbus 软元件偏置值
	GD61020	运行数据输入画面的运行号码的 No. 显示
	GD61021	运行数据输入画面的运行方式
	GD61022	运行数据输入画面的位置
	GD61024	运行数据输入画面的运行速度
	GD61026	运行数据输入画面的运行功能
	GD61027	运行数据输入画面的压推电流
	GD61029	运行数据输入画面的停留时间
	GD61031	运行数据输入画面的顺序定位
	GD61032	运行数据输入画面的加速
	GD61034	运行数据输入画面的减速
	GD61040	IN 输入的信号号码
	GD61041	OUT 输出的信号号码
	GD61042	NET-IN 输入的信号号码
	GD61043	NET-OUT 输出的信号号码
	GD61044	IN 输入功能选择的暂存软元件
	GD61046	OUT/NET-OUT 输出功能选择的暂存软元件
	GD61048	NET-IN 输入功能选择的暂存软元件
	GD61050～GD61080	NET-IN0～NET-IN15 的备份软元件
	GD61082～GD61096	IN0～IN7 的备份软元件
	GD61098～GD61108	OUT0～OUT5 的备份软元件
	GD61110	监视的指令位置的备份软元件
	GD61112	预置位置的备份软元件
	GD61114	JOG 移动量的备份软元件(使用 GOT 的最小移动量)
	GD61116	Configuration 执行等待时间用类型
	GD61199	站号备份
	GD61200+100n (n=0～30)	运行号码的 No. 显示(1 轴～31 轴)
	GD61202+100n (n=0～30)	NET-IN0 的备份软元件(1 轴～31 轴)
	GD61204+100n (n=0～30)	NET-IN1 的备份软元件(1 轴～31 轴)
	GD61206+100n (n=0～30)	NET-IN2 的备份软元件(1 轴～31 轴)
	GD61208+100n (n=0～30)	NET-IN3 的备份软元件(1 轴～31 轴)
	GD61210+100n (n=0～30)	NET-IN4 的备份软元件(1 轴～31 轴)
	GD61212+100n (n=0～30)	NET-IN5 的备份软元件(1 轴～31 轴)
	GD61214+100n (n=0～30)	NET-IN6 的备份软元件(1 轴～31 轴)
	GD61216+100n (n=0～30)	NET-IN7 的备份软元件(1 轴～31 轴)
	GD61218+100n (n=0～30)	NET-IN8 的备份软元件(1 轴～31 轴)
	GD61220+100n (n=0～30)	NET-IN9 的备份软元件(1 轴～31 轴)
	GD61222+100n (n=0～30)	NET-IN10 的备份软元件(1 轴～31 轴)
	GD61224+100n (n=0～30)	NET-IN11 的备份软元件(1 轴～31 轴)
	GD61226+100n (n=0～30)	NET-IN12 的备份软元件(1 轴～31 轴)
	GD61228+100n (n=0～30)	NET-IN13 的备份软元件(1 轴～31 轴)
	GD61230+100n (n=0～30)	NET-IN14 的备份软元件(1 轴～31 轴)
	GD61232+100n (n=0～30)	NET-IN15 的备份软元件(1 轴～31 轴)
	GD61234+100n (n=0～30)	IN0 的备份软元件(1 轴～31 轴)
	GD61236+100n (n=0～30)	IN1 的备份软元件(1 轴～31 轴)
	GD61238+100n (n=0～30)	IN2 的备份软元件(1 轴～31 轴)
	GD61240+100n (n=0～30)	IN3 的备份软元件(1 轴～31 轴)

类型	软元件编号	用途
字	GD61242+100n (n=0~30)	IN4 的备份软元件 (1 轴~31 轴)
	GD61244+100n (n=0~30)	IN5 的备份软元件 (1 轴~31 轴)
	GD61246+100n (n=0~30)	IN6 的备份软元件 (1 轴~31 轴)
	GD61248+100n (n=0~30)	IN7 的备份软元件 (1 轴~31 轴)
	GD61250+100n (n=0~30)	监视的指令位置的备份软元件 (1 轴~31 轴)
	GD61252+100n (n=0~30)	预置位置的备份软元件 (1 轴~31 轴)
	GD61254+100n (n=0~30)	JOG 移动量的备份软元件 (1 轴~31 轴)
	GD61256+100n (n=0~30)	Configuration 执行等待时间用计时器 (1 轴~31 轴)
	GD64990~GD64995	时钟的数字开关
	GD65000	号机号码
	GD65001	电源类型指定
	GD65100~GD65102	外部控制软元件 (配方)
	GD65103~GD65105	外部控制软元件 (配方)
	GS7	1 秒二进制计数器
	GS513~GS516	更改时间
	GS531~GS532	监视站断开 (CH1)
	GS570~GS576	MODBUS 通讯控制功能
	GS650~GS652	当前时间
	TMP950~TMP996	脚本运算用

## 5.5 注释一览表

注释组号	注释号	使用处
497	No. 1~240	B-31015、B-31018~B-31019
498	No. 1~60	B-31012、B-31013、B-31016、B-31017、B-31020、W-32004、W-32005
499	No. 1~90	B-31012、B-31013、B-31016、B-31017、B-31020、W-32006
500	No. 1、2	B-30002~B-30006、B-30500~B-31030
	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. 7	B-30002
	No. 8	B-30001
	No. 9~13	B-30003、B-31004~B-31008
	No. 14~16、18	B-30004、B-31010~B-31014
	No. 17	B-30004
	No. 19~20	B-30005、B-30500、B-31015~B-31019
	No. 21~23	B-30005
	No. 24~26	B-30006、B-31020~B-31029
	No. 27	B-30003~B-30006、B-31002、B-31010~B-31014、B-31030
	No. 28	B-30003~B-30006、B-31002~B-31008、B-31030
	No. 29~31	B-30003~B-30006、B-31002、B-31030
	No. 32~33	B-31004~B-31008
	No. 34	B-31010~B-31014
	No. 35、38~40	B-30500、B-31015~B-31019
	No. 36	B-30003~B-30005、B-30500~B-31019、B-31030
	No. 37	B-30005、B-30500
	No. 50~51	B-30001~B-30006、B-31002~B-31021、B-31030
	No. 100~128	B-31002
	No. 150~185	B-31004
	No. 200~218	B-31005
	No. 250~269	B-31006
	No. 300~316	B-31007



注释组号	注释号	使用处
500	No. 350～362	B-31008
	No. 400～413	B-31010
	No. 450～457	B-31011
	No. 500～524	B-31012
	No. 550～582	B-31013
	No. 600～602	B-31014
	No. 650～660	B-31015
	No. 700～702	B-31016
	No. 750～752	B-31017
	No. 800～806	B-31018
	No. 850～856	B-31019
	No. 900～905	B-31020
	No. 950～965	B-31021
	No. 1000～1014、 1046	B-31022～B-31029
	No. 1015～1018	B-31022
	No. 1019～1022	B-31023
	No. 1023～1026	B-31024
	No. 1027～1030	B-31025
	No. 1031～1034	B-31026
	No. 1035～1038	B-31027
	No. 1039～1042	B-31028
	No. 1043～1045	B-31029
	No. 1050～1064	B-31030
	No. 1100～1101	W-30001
	No. 1150	W-30002
	No. 1200～1208	W-30003
	No. 1250～1284	W-30004
	No. 1300～1334	W-32001
	No. 1350～1354	W-32002
	No. 1400～1403	W-32003
	No. 1450～1451	W-32004～W-32006
	No. 1500～1503	W-32007
	No. 1550～1552	W-32008

## 5.6 脚本一览表

项目	设置
工程脚本	有
画面脚本	B-30500、B-31002、B-31021~B-31029、W-30004、W-32001、W-32004~W-32006
对象脚本	B-30500、W-30003

### 5.6.1 工程脚本

脚本号	30001	脚本名	Script30001
注释	初期设置		
数据类型	有符号 BIN16	触发类型	上升沿 GB40
<pre>[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)</pre>			
脚本号	31001	脚本名	Script31001
注释	运行数据画面的 No 的初期值设置		
数据类型	有符号 BIN16	触发类型	上升沿 GB40
<pre>// 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.</pre>			
脚本号	31002	脚本名	Script31002
注释	连接机器的通讯全轴有效		
数据类型	有符号 BIN16	触发类型	周期 1(秒)
<pre>// 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] &gt;= 30001 &amp;&amp; [w:GD60000] &lt;= 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;</pre>			

[w:GS532] = 0; }			
脚本号	31014	脚本名	Script31014
注释	互锁		
数据类型	有符号 BIN16	触发类型	通常
<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 画面脚本

### 基本画面 30500

脚本号	30002	脚本名	Script30002
注释	手册显示画面的语言切换		
数据类型	无符号 BIN16	触发类型	通常
<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>			

### 基本画面 31002

脚本号	31003	脚本名	Script31003
注释	上方的显示		
数据类型	有符号 BIN16	触发类型	上升沿 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</pre>			

<pre> [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>			
脚本号	31004	脚本名	Script31004
注释	下方的显示		
数据类型	有符号 BIN16	触发类型	上升沿 GB62001
<pre> // 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] &gt;= 8) { [w:GD61010] = 7; } // Not display No.63 or higher after the Downward button is clicked 8 times or more. if ([w:GD61011] &gt;= 128) { [w:GD61011] = 112; } // Same as above (Modbus offset 16 multiplied by 7=112 No.56 is the starting point)  [b:GB62001] = OFF; </pre>			
脚本号	31013	脚本名	Script31013
注释	位置确定		
数据类型	有符号 BIN16	触发类型	上升沿 GB62019
<pre> // teaching set position // Script for teaching set positon 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 Min If the command position exceed the specified limit if ([s32:400199] &lt;= -8388608    [s32:400199] &gt;= 8388607) {   if ([s32:400199] &lt;= -8388608) {     [s32:401025[u16:GD61015]] = -8388608; // Clip by means of Min   }   if ([s32:400199] &gt;= 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; </pre>			

脚本号	31005	脚本名	Script31005
注释	输入运行数据_读入		
数据类型	有符号 BIN16	触发类型	ON 中 GB62002
<pre>// 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</pre>			

```

[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

```

## 基本画面 31002、31021

脚本号	31008	脚本名	Script31008
注释	测试运行时指令位置显示用		
数据类型	有符号 BIN16	触发类型	通常
<pre>// 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</pre>			
脚本号	31011	脚本名	Script31011
注释	运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62015
<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).  // 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     } }</pre>			

## 基本画面 31002、31020、31021

脚本号	31009	脚本名	Script31009
注释	启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) 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)</pre>			

```

// 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 && [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
}

```

脚本号	31010	脚本名	Script31010
注释	运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 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
    }
}

```

<pre> [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] &gt; 1 &amp;&amp; [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 } </pre>			
脚本号	31012	脚本名	Script31012
注释	运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62016
<pre> // 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. </pre>			

```

// [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
}

```

### 基本画面 31022～31030

脚本号	31015	脚本名	Script31015
注释	多点运行互锁		
数据类型	有符号 BIN16	触发类型	通常

```

// 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; }

```

### 基本画面 31022

脚本号	31016	脚本名	Script31016
注释	1-4_多点运行启动_结束确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62020
// 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([l-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([l-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 && [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
        [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([l-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([l-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
        }
    }
}

```



<pre>     } 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     } } </pre>			
脚本号	31017	脚本名	Script31017
注释	1-4_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 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 </pre>			

```

[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
[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)
}
```

脚本号	31018	脚本名	Script31018
注释	1-4_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 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
  }
}
}

```

脚本号	31019	脚本名	Script31019
注释	1-4_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 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
}

```

脚本号	31016	脚本名	Script31016
注释	1-4_多点运行启动_结束确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) B62030
基本画面 31022 的脚本与 No. 31016 相同。			
脚本号	31017	脚本名	Script31017
注释	1-4_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62033
基本画面 31022 的脚本与 No. 31017 相同。			
脚本号	31018	脚本名	Script31018
注释	1-4_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62035
基本画面 31022 的脚本与 No. 31018 相同。			
脚本号	31019	脚本名	Script31019
注释	1-4_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62036

基本画面 31022 的脚本与 No. 31019 相同。			
脚本号	31016	脚本名	Script31016
注释	1-4_多点运行启动_结束确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62040
基本画面 31022 的脚本与 No. 31016 相同。			
脚本号	31017	脚本名	Script31017
注释	1-4_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62043
基本画面 31022 的脚本与 No. 31017 相同。			
脚本号	31018	脚本名	Script31018
注释	1-4_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62045
基本画面 31022 的脚本与 No. 31018 相同。			
脚本号	31019	脚本名	Script31019
注释	1-4_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62046
基本画面 31022 的脚本与 No. 31019 相同。			
脚本号	31016	脚本名	Script31016
注释	1-4_多点运行启动_结束确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62050
基本画面 31022 的脚本与 No. 31016 相同。			
脚本号	31017	脚本名	Script31017
注释	1-4_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62053
基本画面 31022 的脚本与 No. 31017 相同。			
脚本号	31018	脚本名	Script31018
注释	1-4_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62055
基本画面 31022 的脚本与 No. 31018 相同。			
脚本号	31019	脚本名	Script31019
注释	1-4_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62056
基本画面 31022 的脚本与 No. 31019 相同。			

### 基本画面 31023

脚本号	31020	脚本名	Script31020
注释	5-8_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62060
<pre>// 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</pre>			

```

// 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([l-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([l-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([l-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([l-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([l-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
}
}

```

脚本号	31021	脚本名	Script31021
注释	5-8_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 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)
}

```

脚本号	31022	脚本名	Script31022
注释	5-8_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 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], [u32GD61700], [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. (400915) 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
    }
}

```

脚本号	31023	脚本名	Script31023
注释	5-8_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62066

```

// 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 && [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] > 1 && [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
// 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
}

```

脚本号	31020	脚本名	Script31020
注释	5-8_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62070
基本画面 31023 的脚本与 No. 31020 相同。			
脚本号	31021	脚本名	Script31021
注释	5-8_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62073
基本画面 31023 的脚本与 No. 31021 相同。			
脚本号	31022	脚本名	Script31022
注释	5-8_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62075



基本画面 31023 的脚本与 No. 31022 相同。			
脚本号	31023	脚本名	Script31023
注释	5-8_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62076
基本画面 31023 的脚本与 No. 31023 相同。			
脚本号	31020	脚本名	Script31020
注释	5-8_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62080
基本画面 31023 的脚本与 No. 31020 相同。			
脚本号	31021	脚本名	Script31021
注释	5-8_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62083
基本画面 31023 的脚本与 No. 31021 相同。			
脚本号	31022	脚本名	Script31022
注释	5-8_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62085
基本画面 31023 的脚本与 No. 31022 相同。			
脚本号	31023	脚本名	Script31023
注释	5-8_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62086
基本画面 31023 的脚本与 No. 31023 相同。			
脚本号	31020	脚本名	Script31020
注释	5-8_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62090
基本画面 31023 的脚本与 No. 31020 相同。			
脚本号	31021	脚本名	Script31021
注释	5-8_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62093
基本画面 31023 的脚本与 No. 31021 相同。			
脚本号	31022	脚本名	Script31022
注释	5-8_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62095
基本画面 31023 的脚本与 No. 31022 相同。			
脚本号	31023	脚本名	Script31023
注释	5-8_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62096
基本画面 31023 的脚本与 No. 31023 相同。			

#### 基本画面 31024

脚本号	31024	脚本名	Script31024
注释	9-12_多点运行启动_结束前确认		

数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62100
<pre>// 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 &amp;&amp; [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         }     } }</pre>			

```

        [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([l-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([l-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([l-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
}
}
```

脚本号	31025	脚本名	Script31025
注释	9-12_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 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)
}

```

脚本号	31026	脚本名	Script31026
注释	9-12_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 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
    }
}
}

```

脚本号	31027	脚本名	Script31027
注释	9-12_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 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
}

```

脚本号	31024	脚本名	Script31024
注释	9-12_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62110

基本画面 31024 的脚本与 No. 31024 相同。

脚本号	31025	脚本名	Script31025
注释	9-12_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62113
基本画面 31024 的脚本与 No. 31025 相同。			
脚本号	31026	脚本名	Script31026
注释	9-12_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62115
基本画面 31024 的脚本与 No. 31026 相同。			
脚本号	31027	脚本名	Script31027
注释	9-12_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62116
基本画面 31024 的脚本与 No. 31027 相同。			
脚本号	31024	脚本名	Script31024
注释	9-12_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62120
基本画面 31024 的脚本与 No. 31024 相同。			
脚本号	31025	脚本名	Script31025
注释	9-12_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62123
基本画面 31024 的脚本与 No. 31025 相同。			
脚本号	31026	脚本名	Script31026
注释	9-12_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62125
基本画面 31024 的脚本与 No. 31026 相同。			
脚本号	31027	脚本名	Script31027
注释	9-12_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62126
基本画面 31024 的脚本与 No. 31027 相同。			
脚本号	31024	脚本名	Script31024
注释	9-12_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62130
基本画面 31024 的脚本与 No. 31024 相同。			
脚本号	31025	脚本名	Script31025
注释	9-12_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62133
基本画面 31024 的脚本与 No. 31025 相同。			
脚本号	31026	脚本名	Script31026
注释	9-12_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62135
基本画面 31024 的脚本与 No. 31026 相同。			
脚本号	31027	脚本名	Script31027

注释	9-12_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62136
基本画面 31024 的脚本与 No. 31027 相同。			



## 基本画面 31025

脚本号	31028	脚本名	Script31028
注释	13-16_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) 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([l-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([l-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.     }      // Distinguish Operation start and Operation-end after finishing the check mentioned above     if([b:GB62141] == 1 &amp;&amp; [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         }     } }</pre>			

```

    } else {
        [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([l-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([l-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.
る
        [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
}
}

```

脚本号	31029	脚本名	Script31029
注释	13-16_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 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)
}

```

脚本号	31030	脚本名	Script31030
注释	13-16_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 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], [u32GD62500], [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
    }
}

```

脚本号	31031	脚本名	Script31031
注释	13-16_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 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-INO-15 from backup
    bmov([u32:GD62734], [1-16:u32:404353], 8); // Restore INO-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
}

```

}			
脚本号	31028	脚本名	Script31028
注释	13-16_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62150
基本画面 31025 的脚本与 No. 31028 相同。			
脚本号	31029	脚本名	Script31029
注释	13-16_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62153
基本画面 31025 的脚本与 No. 31029 相同。			
脚本号	31030	脚本名	Script31030
注释	13-16_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62155
基本画面 31025 的脚本与 No. 31030 相同。			
脚本号	31031	脚本名	Script31031
注释	13-16_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62156
基本画面 31025 的脚本与 No. 31031 相同。			
脚本号	31028	脚本名	Script31028
注释	13-16_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62160
基本画面 31025 的脚本与 No. 31028 相同。			
脚本号	31029	脚本名	Script31029
注释	13-16_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62163
基本画面 31025 的脚本与 No. 31029 相同。			
脚本号	31030	脚本名	Script31030
注释	13-16_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62165
基本画面 31025 的脚本与 No. 31030 相同。			
脚本号	31031	脚本名	Script31031
注释	13-16_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62166
基本画面 31025 的脚本与 No. 31031 相同。			
脚本号	31028	脚本名	Script31028
注释	13-16_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62170
基本画面 31025 的脚本与 No. 31028 相同。			
脚本号	31029	脚本名	Script31029
注释	13-16_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62173

基本画面 31025 的脚本与 No. 31029 相同。			
脚本号	31030	脚本名	Script31030
注释	13-16_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62175
基本画面 31025 的脚本与 No. 31030 相同。			
脚本号	31031	脚本名	Script31031
注释	13-16_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62176
基本画面 31025 的脚本与 No. 31031 相同。			

### 基本画面 31026

脚本号	31032	脚本名	Script31032
注释	17-20_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62180
<pre>// 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)</pre>			

```

} else {
    [b:GB62182] = 0;      // In case the motor is running
    [w:GD60004] = 32008;  // In operation checking flag (0: in operation)
    // 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([l-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([l-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
}
}

```

脚本号	31033	脚本名	Script31033
注释	17-20_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 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-INO-15
// [u32:GD62834], [u32:GD62934], [u32:GD63034], [u32:GD63134] : Backup device for INO-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)
}

```

脚本号	31034	脚本名	Script31034
注释	17-20_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 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
    }
}

```



<pre> [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 } } </pre>			
脚本号	31035	脚本名	Script31035
注释	17-20_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62186
<pre> // 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 &amp;&amp; [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] &gt; 1 &amp;&amp; [b:GB62187] == 1){ </pre>			

```

[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
}
```

脚本号	31032	脚本名	Script31032
注释	17-20_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62190
基本画面 31026 的脚本与 No. 31032 相同。			
脚本号	31033	脚本名	Script31033
注释	17-20_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62193
基本画面 31026 的脚本与 No. 31033 相同。			
脚本号	31034	脚本名	Script31034
注释	17-20_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62195
基本画面 31026 的脚本与 No. 31034 相同。			
脚本号	31035	脚本名	Script31035
注释	17-20_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62196
基本画面 31026 的脚本与 No. 31035 相同。			
脚本号	31032	脚本名	Script31032
注释	17-20_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62200
基本画面 31026 的脚本与 No. 31032 相同。			
脚本号	31033	脚本名	Script31033
注释	17-20_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62203
基本画面 31026 的脚本与 No. 31033 相同。			
脚本号	31034	脚本名	Script31034
注释	17-20_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62205
基本画面 31026 的脚本与 No. 31034 相同。			
脚本号	31035	脚本名	Script31035
注释	17-20_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62206
基本画面 31026 的脚本与 No. 31035 相同。			

脚本号	31032	脚本名	Script31032
注释	17-20_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62210
基本画面 31026 的脚本与 No. 31032 相同。			
脚本号	31033	脚本名	Script31033
注释	17-20_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62213
基本画面 31026 的脚本与 No. 31033 相同。			
脚本号	31034	脚本名	Script31034
注释	17-20_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62215
基本画面 31026 的脚本与 No. 31034 相同。			
脚本号	31035	脚本名	Script31035
注释	17-20_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62216
基本画面 31026 的脚本与 No. 31035 相同。			

### 基本画面 31027

脚本号	31036	脚本名	Script31036
注释	21-24_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62220
<pre>// 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([l-21:s32:400129] == 0){ // In case of no alarm</pre>			

```

    [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 {              // Prepare Operation-end in case the touch panel is in operation.
        [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([l-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([l-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([l-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([l-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
    }
}
}

```

脚本号	31037	脚本名	Script31037
注释	21-24_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 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)
}

```

脚本号	31038	脚本名	Script31038
注释	21-24_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 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
    }
}

```

脚本号	31039	脚本名	Script31039
注释	21-24 多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 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
}
```

脚本号	31036	脚本名	Script31036
注释	21-24_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62230
基本画面 31027 的脚本与 No. 31036 相同。			

脚本号	31037	脚本名	Script31037
注释	21-24_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62233
基本画面 31027 的脚本与 No. 31037 相同。			

脚本号	31038	脚本名	Script31038
注释	21-24_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62235
基本画面 31027 的脚本与 No. 31038 相同。			

脚本号	31039	脚本名	Script31039
注释	21-24_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62236
基本画面 31027 的脚本与 No. 31039 相同。			

脚本号	31036	脚本名	Script31036
注释	21-24_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62240
基本画面 31027 的脚本与 No. 31036 相同。			

脚本号	31037	脚本名	Script31037
注释	21-24_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62243
基本画面 31027 的脚本与 No. 31037 相同。			

脚本号	31038	脚本名	Script31038
注释	21-24_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62245
基本画面 31027 的脚本与 No. 31038 相同。			
脚本号	31039	脚本名	Script31039
注释	21-24_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62246
基本画面 31027 的脚本与 No. 31039 相同。			
脚本号	31036	脚本名	Script31036
注释	21-24_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62250
基本画面 31027 的脚本与 No. 31036 相同。			
脚本号	31037	脚本名	Script31037
注释	21-24_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62253
基本画面 31027 的脚本与 No. 31037 相同。			
脚本号	31038	脚本名	Script31038
注释	21-24_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62255
基本画面 31027 的脚本与 No. 31038 相同。			
脚本号	31039	脚本名	Script31039
注释	21-24_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62256
基本画面 31027 的脚本与 No. 31039 相同。			

### 基本画面 31028

脚本号	31040	脚本名	Script31040
注释	25-28_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62260
<pre>// Script to check whether an alarm is not occurring or Operation is not running before multiple operation start/end // Screen script for 25-28 axis multiple operation  // 25-28_Check before start/end multiple operation script No.31040 // Start the script with the respective [Start] button (2 seconds ON cycle) in the multiple operation screen of 25-28 axis  // [b:GB62260] : 25 axis starting trigger using bit set // [b:GB62270] : 26 axis starting trigger using bit set // [b:GB62280] : 27 axis starting trigger using bit set // [b:GB62290] : 28 axis starting trigger using bit set // [b:GB62265] : 25 axis touch panel in operation flag // [b:GB62275] : 26 axis touch panel in operation flag // [b:GB62285] : 27 axis touch panel in operation flag // [b:GB62295] : 28 axis touch panel in operation flag  // [w:GD61000] = 1 means as below</pre>			

```

// 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

// 25 axis
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([l-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([l-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
    }
}
}

```

脚本号	31041	脚本名	Script31041
注释	25-28_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62263
<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 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 &amp;&amp; [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</pre>			

```

[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

```

<pre> [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) } </pre>			
脚本号	31042	脚本名	Script31042
注释	25-28_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 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], [u32GD63700], [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     } } } </pre>			

```

// 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
    }
}

// 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
    }
}

```

脚本号	31043	脚本名	Script31043
注释	25-28_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 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
}

```

脚本号	31040	脚本名	Script31040
注释	25-28_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62270
基本画面 31028 的脚本与 No. 31040 相同。			
脚本号	31041	脚本名	Script31041
注释	25-28_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62273
基本画面 31028 的脚本与 No. 31041 相同。			
脚本号	31042	脚本名	Script31042
注释	25-28_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62275
基本画面 31028 的脚本与 No. 31042 相同。			
脚本号	31043	脚本名	Script31043
注释	25-28_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62276
基本画面 31028 的脚本与 No. 31043 相同。			
脚本号	31040	脚本名	Script31040



注释	25-28_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62280
基本画面 31028 的脚本与 No. 31040 相同。			
脚本号	31041	脚本名	Script31041
注释	25-28_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62283
基本画面 31028 的脚本与 No. 31041 相同。			
脚本号	31042	脚本名	Script31042
注释	25-28_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62285
基本画面 31028 的脚本与 No. 31042 相同。			
脚本号	31043	脚本名	Script31043
注释	25-28_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62286
基本画面 31028 的脚本与 No. 31043 相同。			
脚本号	31040	脚本名	Script31040
注释	25-28_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62290
基本画面 31028 的脚本与 No. 31040 相同。			
脚本号	31041	脚本名	Script31041
注释	25-28_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62293
基本画面 31028 的脚本与 No. 31041 相同。			
脚本号	31042	脚本名	Script31042
注释	25-28_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62295
基本画面 31028 的脚本与 No. 31042 相同。			
脚本号	31043	脚本名	Script31043
注释	25-28_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62296
基本画面 31028 的脚本与 No. 31043 相同。			

### 基本画面 31029

脚本号	31044	脚本名	Script31044
注释	29-31_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) 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</pre>			

```

// [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([l-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([l-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 && [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
    }
}

// 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
}
}

```

脚本号	31045	脚本名	Script31045
注释	29-31_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 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)
}

```

脚本号	31046	脚本名	Script31046
注释	29-31_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 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
    }
}

```

脚本号	31047	脚本名	Script31047
注释	29-31_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 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
}

```

脚本号	31044	脚本名	Script31044
注释	29-31_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62310
基本画面 31029 的脚本与 No. 31044 相同。			
脚本号	31045	脚本名	Script31045
注释	29-31_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62313

基本画面 31029 的脚本与 No. 31045 相同。			
脚本号	31046	脚本名	Script31046
注释	29-31_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62315
基本画面 31029 的脚本与 No. 31046 相同。			
脚本号	31047	脚本名	Script31047
注释	29-31_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62316
基本画面 31029 的脚本与 No. 31047 相同。			
脚本号	31044	脚本名	Script31044
注释	29-31_多点运行启动_结束前确认		
数据类型	有符号 BIN16	触发类型	ON 中周期 2(秒) GB62320
基本画面 31029 的脚本与 No. 31044 相同。			
脚本号	31045	脚本名	Script31045
注释	29-31_多点运行准备		
数据类型	有符号 BIN16	触发类型	ON 中 GB62323
基本画面 31029 的脚本与 No. 31045 相同。			
脚本号	31046	脚本名	Script31046
注释	29-31_多点运行中		
数据类型	有符号 BIN16	触发类型	ON 中 GB62325
基本画面 31029 的脚本与 No. 31046 相同。			
脚本号	31047	脚本名	Script31047
注释	29-31_多点运行结束		
数据类型	有符号 BIN16	触发类型	ON 中 GB62326
基本画面 31029 的脚本与 No. 31047 相同。			

#### 窗口画面 30004

脚本号	30100	脚本名	Script30100
注释	轴切换		
数据类型	无符号 BIN16	触发类型	关闭画面时
<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]);</pre>			

```
rst([b:GB61001]);
rst([b:GB61002]);
```

### 窗口画面 32001

脚本号	31006	脚本名	Script31006
注释	输入运行数据_写入		
数据类型	有符号 BIN16	触发类型	ON 中 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 // 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</pre>			

```

[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

```

# 窗口画面 32004～32006

脚本号	31007	脚本名	Script31007
注释	IN_OUT 功能选择_写入		
数据类型	有符号 BIN16	触发类型	ON 中 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             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     } }</pre>			

```

        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 对象脚本

#### 基本画面 30500

对象(名称)	开关	对象 ID *1	20035
脚本用户 ID	1		
数据类型	无符号 BIN16	触发类型	软元件写入时
//Prevents exceeding the total number of the document pages. if([u16:GD60081] >= [u16:GD60082]){ [u16:GD60081] = [u16:GD60082] - 1; } 			

#### 窗口画面 30003

对象	数据显示	对象 ID *1	10014
脚本用户 ID	1		
数据类型	无符号 BIN16	触发类型	上升沿 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

```

对象	数据显示	对象 ID *1	10015
脚本用户 ID	2		
数据类型	无符号 BIN16	触发类型	通常

//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

```

对象	数据显示	对象 ID *1	10016
脚本用户 ID	3		
数据类型	无符号 BIN16	触发类型	通常

//Year & Month Setting

```

[w:GS513] = ([w:TMP980] << 8) + [w:TMP981]; //Set Year & Month to Change Time Device

```

对象	数据显示	对象 ID *1	10017
脚本用户 ID	4		
数据类型	无符号 BIN16	触发类型	通常
//Date & Time Setting			
[w:GS514] = ([w:TMP982] << 8) + [w:TMP983]; //Set Date & Time to Change Time Device			
对象	数据显示	对象 ID *1	10018
脚本用户 ID	5		
数据类型	无符号 BIN16	触发类型	通常
//Minute & Second Setting			
[w:GS515] = ([w:TMP984] << 8) + [w:TMP985]; //Set Minute & Second to Change Time Device			
对象	数据显示	对象 ID *1	10019
脚本用户 ID	6		
数据类型	无符号 BIN16	触发类型	通常
//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 对象 ID 引用画面时有可能被变更。

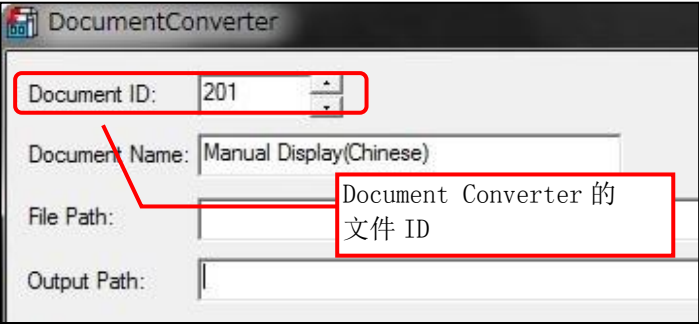
6. 关于手册显示

手册显示使用文件显示功能显示。关于文件显示功能的详细内容, 请参照「GT Designer3 (GOT2000) 帮助」。文件显示功能本身并不能切换语言, 所以在样本画面中, 通过所选的显示语言变更文件 ID, 将实现文件的语言切换。

6.1 手册显示用文件数据的准备

例: 基本画面 B-30500: 在手册显示中, 显示中文(简体)手册(文件)时

- (1) 使用 Document Converter, 将显示手册(Word、Excel 等)转换为文件显示功能用的文件数据(JPEG 文件)。在 Document Converter 的[文件 ID]中设置 201。  
※文件 ID 和显示语言对应, 请参照如下表。



注释组号	语言	文件 ID 列号
1	中文(简体)	201
2	日语	202
3	英语	203

※请使用 2.09K 以后的 Document Converter 版本。如 2.08J 以前版本的话, 切换总页数和页数的开关不能正确地动作。

- (2) 在 DOCIMG 文件夹的 201 文件夹中生成文件数据。不更改 DOCIMG 文件夹以下的文件夹构成, 将整个 DOCIMG 文件夹一并保存在 SD 卡的根目录中。



SD 卡的文件夹构成

备注: 总页数 100 页以上时  
该样品的总页数设定为 99 页的文件。如果超过了 100 页, 请修改该总页数以及进行显示当前页号码的该数值显示的格式字符串(# 的个数)。

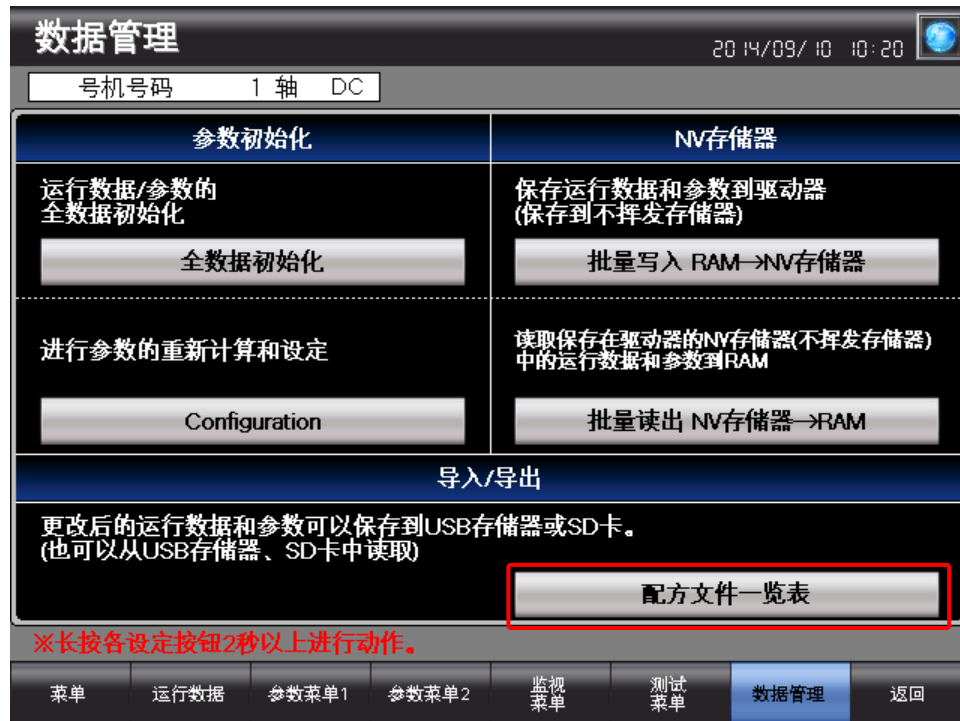
## 7. 关于对 USB 存储器、SD 卡存取运行数据及参数

通过使用配方文件，运行数据及参数既可以保存至 USB 存储器或 SD 卡，也可以从 USB 存储器或 SD 卡读取。

### 7.1 运行数据及参数保存至 USB 存储器或 SD 卡的方法

在此举例说明如何将数据保存至 USB 存储器。

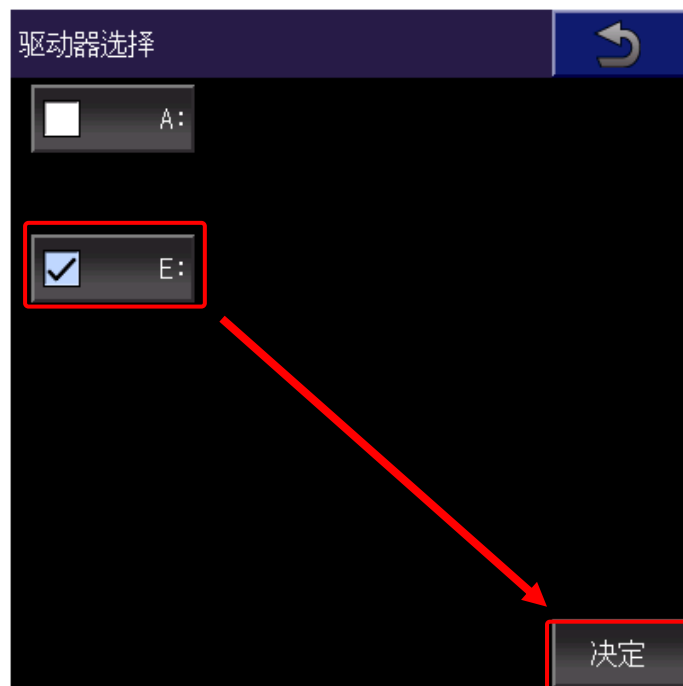
(1) 触摸数据管理画面的「配方文件一览表」。



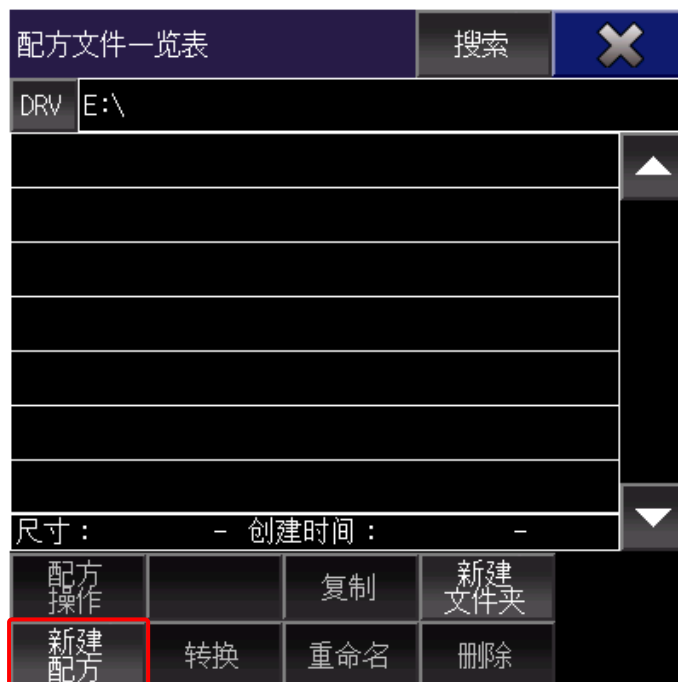
(2) 显示配方文件一览表的窗口。请触摸窗口中的「DRV」。



- (3) 显示驱动器选择。  
请触摸 E 驱动器。  
然后，请触摸「决定」。



- (4) 请触摸新建配方。



(5) 触摸新建配方后，即显示配方设置名(配方名)一览表。如下图所示。

新建配方		搜索	
号	配方设置名		
30001	No.1	驱动器	运行数据
30002	No.1	驱动器	参数 AC
30003	No.1	驱动器	参数 DC
30004	No.2	驱动器	运行数据
30005	No.2	驱动器	参数 AC
30006	No.2	驱动器	参数 DC
30007	No.3	驱动器	运行数据
30008	No.3	驱动器	参数 AC
		文件生成	

配方名的含义，如下所示。

号	配方设置名		
30001	No.1	驱动器	运行数据
30002	No.1	驱动器	参数 AC
30003	No.1	驱动器	参数 DC
30004	No.2	驱动器	运行数据
30005	No.2	驱动器	参数 AC
30006	No.2	驱动器	参数 DC
30007	No.3	驱动器	运行数据
30008	No.3	驱动器	参数 AC

运行数据。

参数。  
分别为 AC 电源和 DC 电源。

No. 的数字，为驱动器的号机号码。  
选择希望保存到 USB 存储器的驱动器的号机号码。

(6) 选择希望保存到 USB 存储器的配方名(各驱动器号机号码的运行数据或参数)。选择后，触摸「文件生成」。

新建配方

搜索

号

配方设置名

30001 No.1 驱动器 运行数据

30002 No.1 驱动器 参数 AC

30003 No.1 驱动器 参数 DC

30004 No.2 驱动器 运行数据

30005 No.2 驱动器 参数 AC

30006 No.2 驱动器 参数 DC

30007 No.3 驱动器 运行数据

30008 No.3 驱动器 参数 AC

文件生成

选择配方名，触摸「文件生成」。

文件名

请输入创建的配方文件名。

TEST

a-z

0-9

@

A B C D E F G

H I J K L M N

O P Q R S T U

V W X Y Z . , SP

AC

DEL

Enter

决定

触摸输入区域，设置配方文件名。  
「Enter」确定输入内容。

文件名

请输入创建的配方文件名。

处理结果

读取配方，  
新的配方文件生成结束。  
TEST.G2P

OK

决定

触摸「决定」，即生成配方文件。



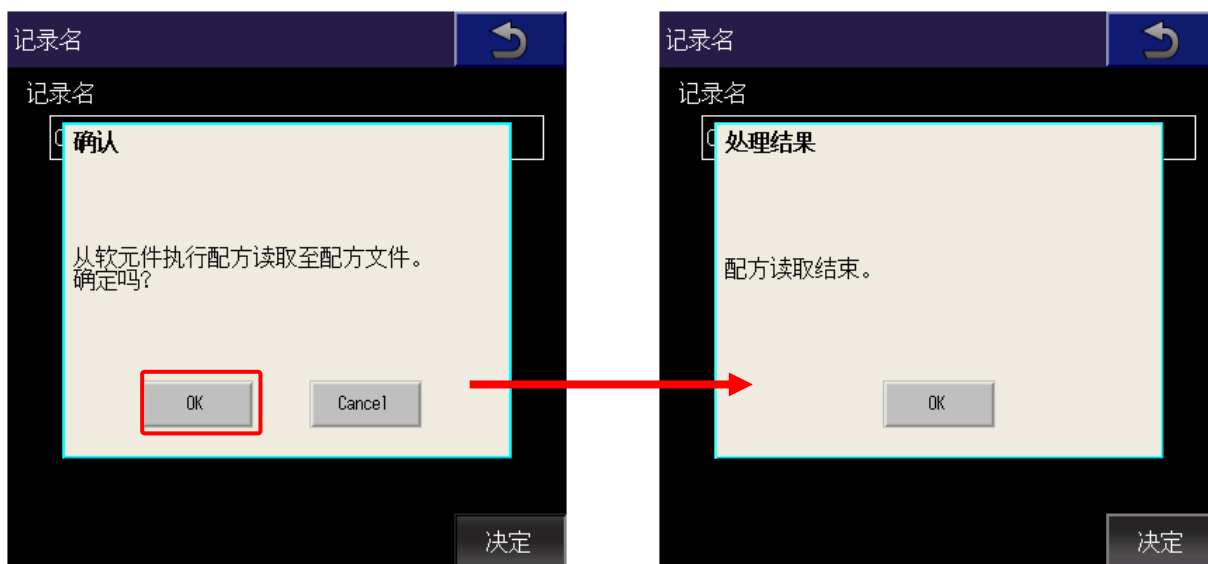
- (7) 生成配方文件后，即显示记录名一览表。如下图所示。  
在此选择「Operation Data」(运行数据)。  
选择「Operation Data」(运行数据)后，触摸「软元件->GOT」。



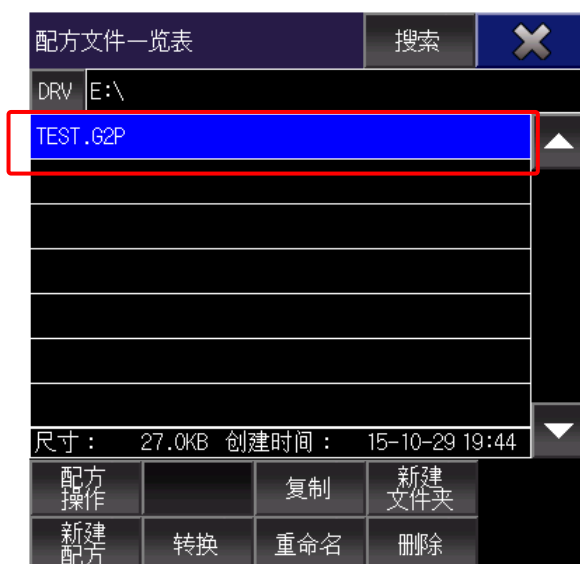
- (8) 如下图所示，显示记录名设置画面。  
如果无需更改记录名，就此触摸「决定」。



(9) 触摸「OK」后，执行配方读取。

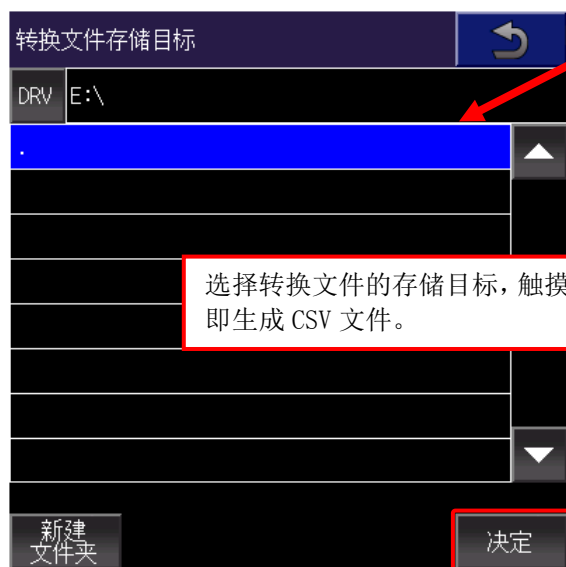
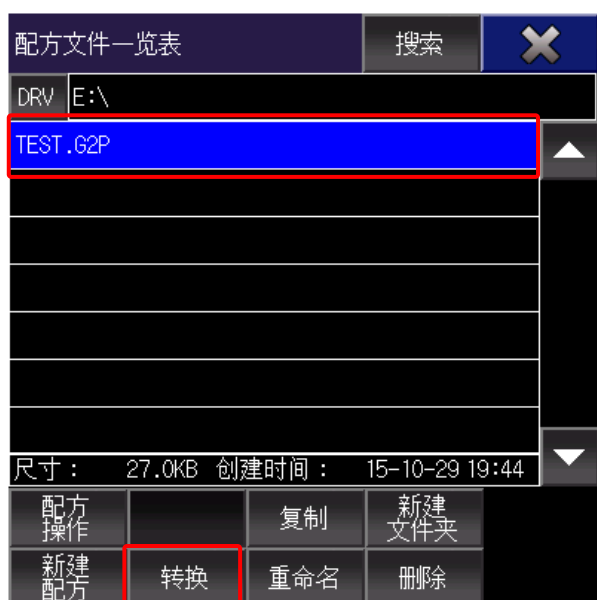


(10) 返回至配方文件一览表画面，可确认 USB 存储器中已经有了从驱动器读取运行数据后生成的配方文件。

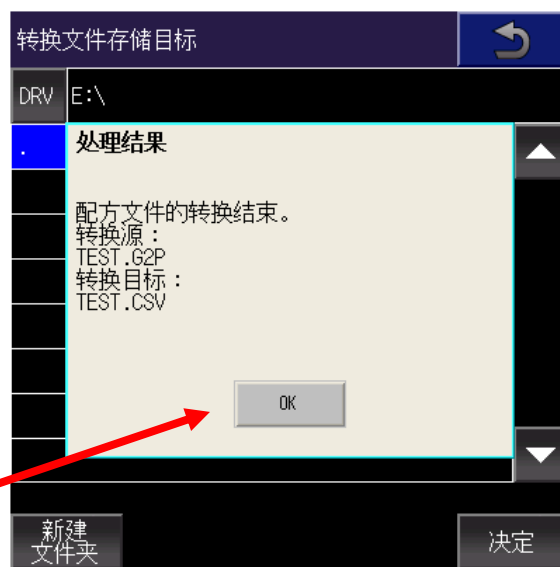


\*参考信息：如果希望保存驱动器(AC 电源)的参数，在(6)中选择「驱动器 参数 AC」后，在(7)中选择「Parameter Data AC」。

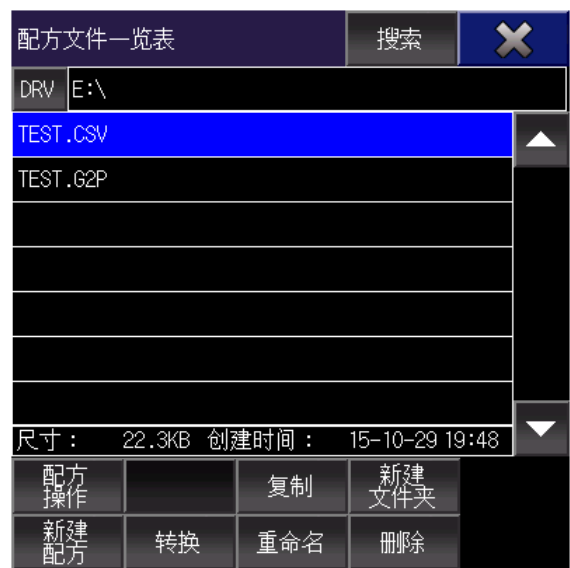
- (11) 为了在计算机中编辑生成的配方文件，将其转换为 CSV 文件。  
选择希望转换的配方文件后，触摸「转换」。



选择转换文件的存储目标，触摸「决定」，  
即生成 CSV 文件。



(12) 通过上述操作，创建如下图所示的 CSV 文件。

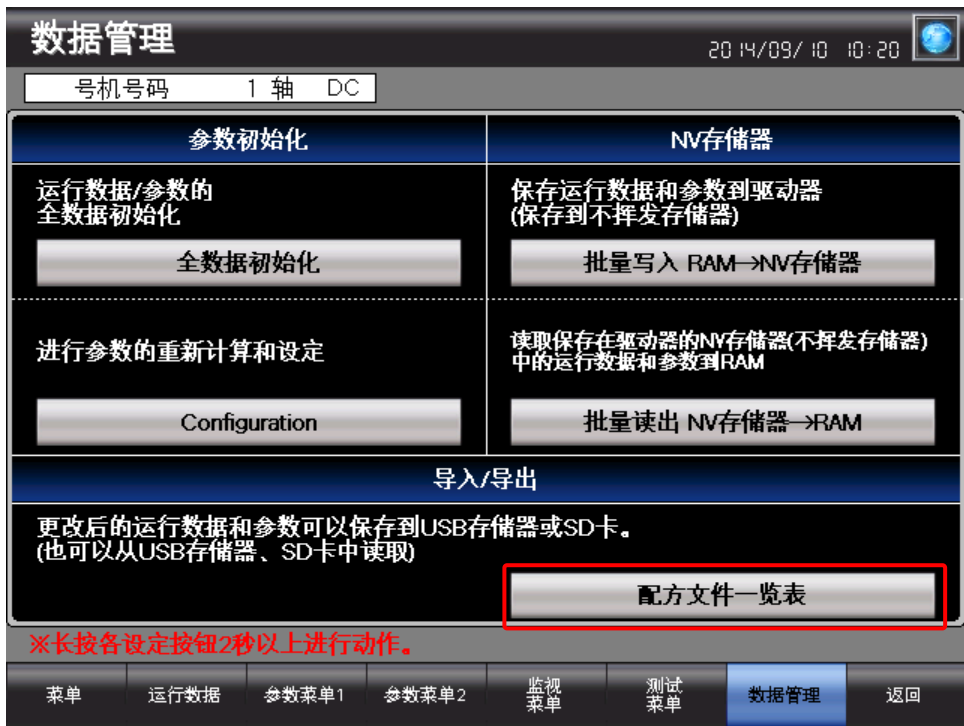


通过上述操作，可将驱动器的运行数据及参数保存至 USB 存储器。

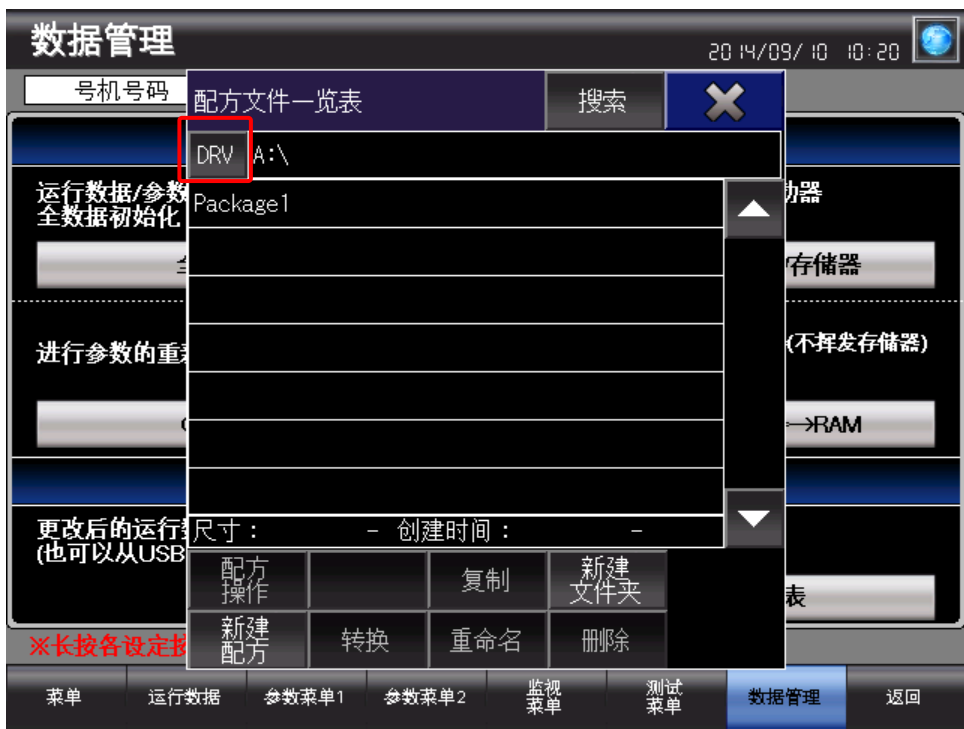
7.2 从 USB 存储器或 SD 卡读取运行数据及参数的方法

在此举例说明如何将保存在 USB 存储器里的运行数据写入驱动器。

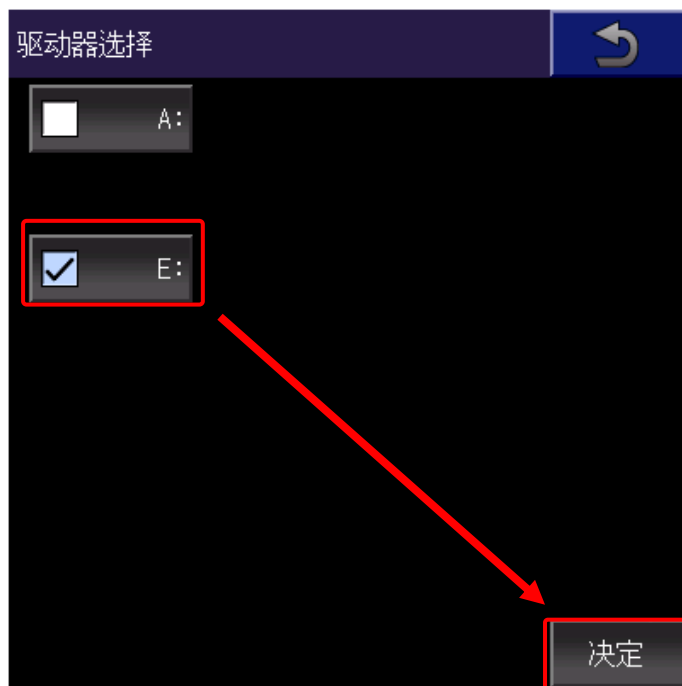
(1) 触摸数据管理画面的「配方文件一览表」。



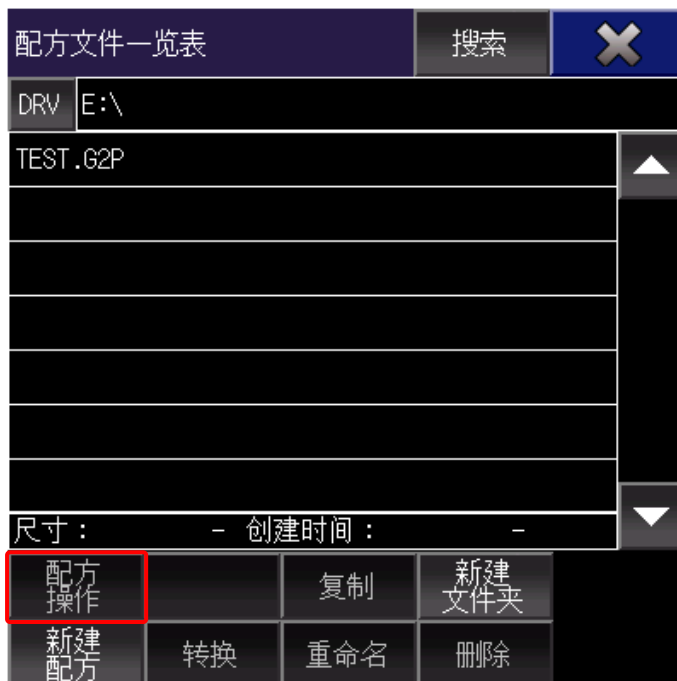
(2) 显示配方文件一览表的窗口。请触摸窗口中的「DRV」。



- (3) 显示驱动器选择。  
请触摸 E 驱动器。  
然后，请触摸「决定」。



- (4) 选择要写入的配方文件后，请触摸「配方操作」。



(5) 选择记录名后，触摸「GOT->软元件」。

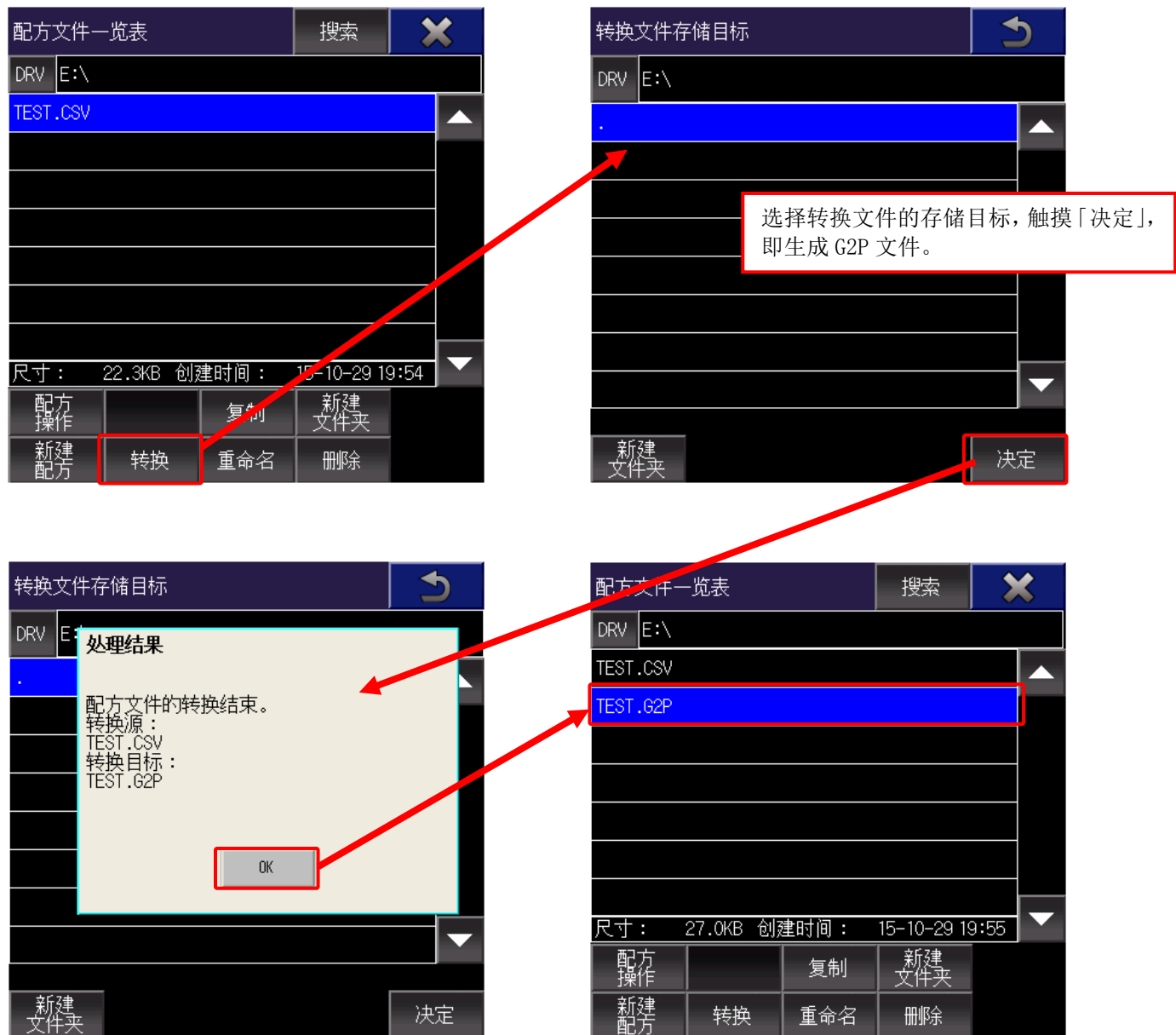


(6) 触摸「OK」后，执行配方写入。



通过上述操作，可将保存在 USB 存储器里的运行数据写入驱动器。

- (7) 为了在配方操作中使用在计算机中编辑过的 CSV 文件，将其转换为 G2P 文件。  
选择希望转换的 CSV 文件，触摸「转换」。



通过上述操作，可将保存在 USB 存储器里的 CSV 文件转换为 G2P 文件。



## 8. 关于 GOT 上的 MODBUS 地址的指定方法

对于 GOT (GT Designer3) 中所使用的软元件，请根据所使用的 MODBUS 机器的地址变换替换为 GT Designer3 中的软元件号后使用。关于 MODBUS 地址的详细内容，请参照「GOT2000 系列 连接手册 (微型计算机/ MODBUS/ 周边机器连接篇)」。

### 8.1 地址的替换方法

要对保持寄存器的地址「1234H」进行监视时，保持寄存器在 GT Designer3 中的地址为「4\*\*\*\*\*」。由于 GT Designer3 中的地址号使用的是 10 进制数，因此将「1234H」转换成 10 进制数后为「04660」。此外，由于 GT Designer3 中的地址号在保存寄存器的情况下是从「1」开始的，因此地址为上述地址「+1」。因此，保存寄存器的地址「1234H」在 GT Designer3 中为「404661」。

例：显示运行速度 No. 0 的数值时

运行速度 No. 0 的高位的 MODBUS 寄存器地址是 10 进制数「1152」。

\*由于数据类型设置为[无符号 BIN32]，因此指定运行速度 No. 0 的高位地址。

实际在 GOT 的保持寄存器中设置的地址为「1152+1」 = 1153，

而 GOT 的保持寄存器的指定地址为「40000+01153」 = 401153。

