

三菱电机株式会社
能量测定单元 EcoMonitorLight
(型号: EMU4-BD1-MB/EMU4-HD1-MB)、
电子式多用指示器 (型号: ME110SSR-MB)

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.....	4
1 概要	5
2 系统构成	5
3 关于 GOT.....	5
3.1 自动选择的系统应用程序	5
3.2 画面创建软件的连接机器设置	5
3.3 画面创建软件的重叠窗口设定	6
4 关于测定终端	6
4.1 测定终端的通讯设置	6
4.2 测定终端的参数设置	6
5 画面规格	8
5.1 显示语言	8
5.2 画面一览表/切换	8
5.2.1 画面一览表/切换(公共)	8
5.2.2 画面一览表/切换(个别)	9
5.3 画面说明	10
5.3.1 菜单(B-30001)	10
5.3.2 当前值监视(B-30002)	11
5.3.3 电能图表(B-30009)	12
5.3.4 综合电流图表(B-30010)	13
5.3.5 参数设置(B-30020)	14
5.3.6 报警记录(B-30011)	15
5.3.7 报警复位(W-30001)	16
5.3.8 语言设置(W-30002)	17
5.3.9 时钟设置(W-30003)	18
5.3.10 输入确认对话框(W-30014)	19
5.3.11 相线系统(W-30015)	20
5.3.12 传感器类型(W-30016)	21
5.3.13 电流・电流需求 乘比率(W-30017)	22
5.3.14 电压 乘比率(W-30018)	23
5.3.15 功率・功率需求・无功功率 乘比率(W-30019)	24
5.3.16 电能・无功电能 乘比率(W-30020)	25
5.3.17 有效/无效站设置(W-30031)	26
5.3.18 站号选择(W-30030)	27
5.4 使用软元件一览表	28
5.4.1 连接机器的软元件	28
5.4.2 GOT 内部软元件	29
5.5 注释一览表	37
5.6 配方一览表	38
5.6.1 公共设置	38
5.6.2 个别设置	38
5.7 脚本一览表	45
5.7.1 工程脚本	45
5.7.2 画面脚本	49
5.7.3 对象脚本	71
6 关于启动时的系统报警(通讯超时)	93

修订记录

样本画面说明书

修订日期	管理编号*	修订内容
2015/1	BCN-P5999-0473	初版
2015/3	BCN-P5999-0473-A	工程数据改进

* 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
2015/1	mitsubishi_EMU4_V_Ver1_C.GTX	1.126G	初版
2015/3	mitsubishi_EMU4_V_Ver1a_C.GTX	1.126G	追加画面的详细说明

* 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 MODBUS®/RTU (RS-485) 连接 GOT2000 与三菱电机株式会社 测定终端 能量测定单元 EcoMonitorLight: EMU4-BD1-MB/EMU4-HD1-MB (以下简称机种 EMU4) 或电子式多用指示器: ME110SSR-MB (以下简称机种 ME110), 进行电流、功率、电能监视等的样本画面说明书。

2 系统构成

GOT2000

- GT27**-V (640×480)
- 接口: 标准 I/F (RS-422/485) *1
- SD 卡 *2
- 电池 (GT11-50BAT 锂电池) *3

RS-485 电缆 *4

三菱电机株式会社
能量测定单元 EcoMonitorLight
(型号: EMU4-BD1-MB/EMU4-HD1-MB)
接口: 输出端子 *5
或电子式多用指示器
(型号: ME110SSR-MB)
接口: 输出端子 *5

- *1: 需使用终端电阻。关于终端电阻。请参照“GOT2000 系列连接手册 (微型计算机/MODBUS®/周边机器连接篇)”。
- *2: SD 卡用于工程数据/OS 的保存和日志功能、配方功能。
- *3: 电池, 用于时钟数据以及 SRAM 用户区的日志数据/用户报警数据的停电保持。(GOT 中标配电池。)
- *4: 关于电缆的详细内容, 请参照“GOT2000 系列连接手册 (微型计算机/MODBUS®/周边机器连接篇)”。
- *5: 需使用终端电阻。关于终端电阻。请参照各设备的手册。

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	日语	
通讯驱动程序	MODBUS®/RTU		
扩展功能	标准字体		中文(简体)
	轮廓字体	黑体	英数假名
			日语汉字
			中文(简体)汉字
	软元件数据传送		

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)	38400	
数据长度	8 位	
停止位	1 位	
奇偶性	偶数	
重试次数(次)	3	
通讯超时时间(秒)	3	
本站地址	1	设置所连接的测定终端的 MODBUS®地址。在本样本中, 固定为 1 站。
发送延迟时间(ms)	0	
32 位存储顺序	HL 顺序	
函数代码[0F]	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)	2	
输入寄存器读取点数(点)	125	
线圈写入点数(点)	800	

项目	设置值	备注
保持寄存器的写入点数 (点)	2	

3.3 画面创建软件的重叠窗口设定

要关闭基本画面切换时窗口画面，在[画面切换/窗口]的重叠窗口的[详细设置]中把[切换基本画面的同时关闭窗口]设定为有效。

4 关于测定终端

4.1 测定终端的通讯设置

(1) 能量测定单元(EMU 4 机种)

项 目	设置值	备 注
MODBUS®地址	1	(*)
MODBUS®波特率	38400 bps	
MODBUS®奇偶性	Even	Odd: 奇数奇偶性
		Even: 偶数奇偶性
		Non: 无奇偶性
MODBUS®停止位	1 位	

(*)MODBUS®地址固定为 1 站。如果未设置为 1 站，将无法正常工作。

(2) 电子式多用指示器(机种 ME110)

项 目	设置值	备 注
MODBUS®地址	1	(*)
MODBUS®波特率	38.4kbps	
MODBUS®奇偶性	Even	Odd: 奇数奇偶性
		Even: 偶数奇偶性
		Non: 无奇偶性
MODBUS®停止位	1 位	

(*)MODBUS®地址固定为 1 站。如果未设置为 1 站，将无法正常工作。

4.2 测定终端的参数设置

本公司动作确认的时候，设置的值如下。

(1) 能量测定单元(机种 EMU4)

项 目	设置值	备 注
相线系统	3	1: 1P2W
		2: 1P3W
		3: 3P3W
		4: 3P4W
一次电压(线电压)	220V	在“相线系统”为 3P4W 以外时有效
一次电压(相电压)	—	在“相线系统”为 3P4W 时有效
传感器类型	0	0: 直接传感器
		2: 5A 传感器
一次电流	1000x0.1A	
电流需求时限	120s	
功率需求时限	120s	
电流・电流需求 乘比率	-2(*1)	-3: $\times 0.001$
		-2: $\times 0.01$
		-1: $\times 0.1$
		0: $\times 1$
电压 乘比率	-1(*1)	-1: $\times 0.1$
		0: $\times 1$

功率・功率需求・无功功率 乘比率	-2(*1)	-3: $\times 0.001$
		-2: $\times 0.01$
		-1: $\times 0.1$
		0: $\times 1$
		1: $\times 10$
电能・无功电能 乘比率	-1(*1)	-2: $\times 0.01$
		-1: $\times 0.1$
		0: $\times 1$
		1: $\times 10$
频率 乘比率	$\times 0.1$	2: $\times 100$
功率因素 乘比率	$\times 0.1$	

(*1)更改相线系统、一次电压、一次电流时，乘比率将自动改变。

(2) 电子式多用指示器(机种 ME110)

项 目	设置值	备 注
相线系统	3	1: 1P2W
		2: 1P3W(RNT)
		3: 3P3W
		4: 3P4W
		5: 1P3W(RNS)
		6: 3P3W_3CT
一次电压(线电压)	6600V	在“相线系统”为 3P4W 以外时有效
一次电压(相电压)	-	在“相线系统”为 3P4W 时有效
一次电流	50x0.1V	
电流需求时限	10s	
电能需求时限	10s	
CO2 排出系数	1.5	
电费单价	1.3	
电流・电流需求 乘比率	1	1: $\times 0.01$
		2: $\times 0.1$
		3: $\times 1$
		4: $\times 10$
电压 乘比率	3	1: $\times 0.1$
		2: $\times 1$
		3: $\times 10$
		4: $\times 100$
功率・功率需求・无功功率 乘比率	3	1: $\times 0.0001$
		2: $\times 0.001$
		3: $\times 0.01$
		4: $\times 0.1$
		5: $\times 1$
		6: $\times 10$
		7: $\times 100$
电能・无功电能 乘比率	2	1: $\times 0.01$
		2: $\times 0.1$
		3: $\times 1$
		4: $\times 10$
		5: $\times 100$
		6: $\times 1000$
频率 乘比率	$\times 0.1$	
功率因素 乘比率	$\times 0.1$	

5 画面规格

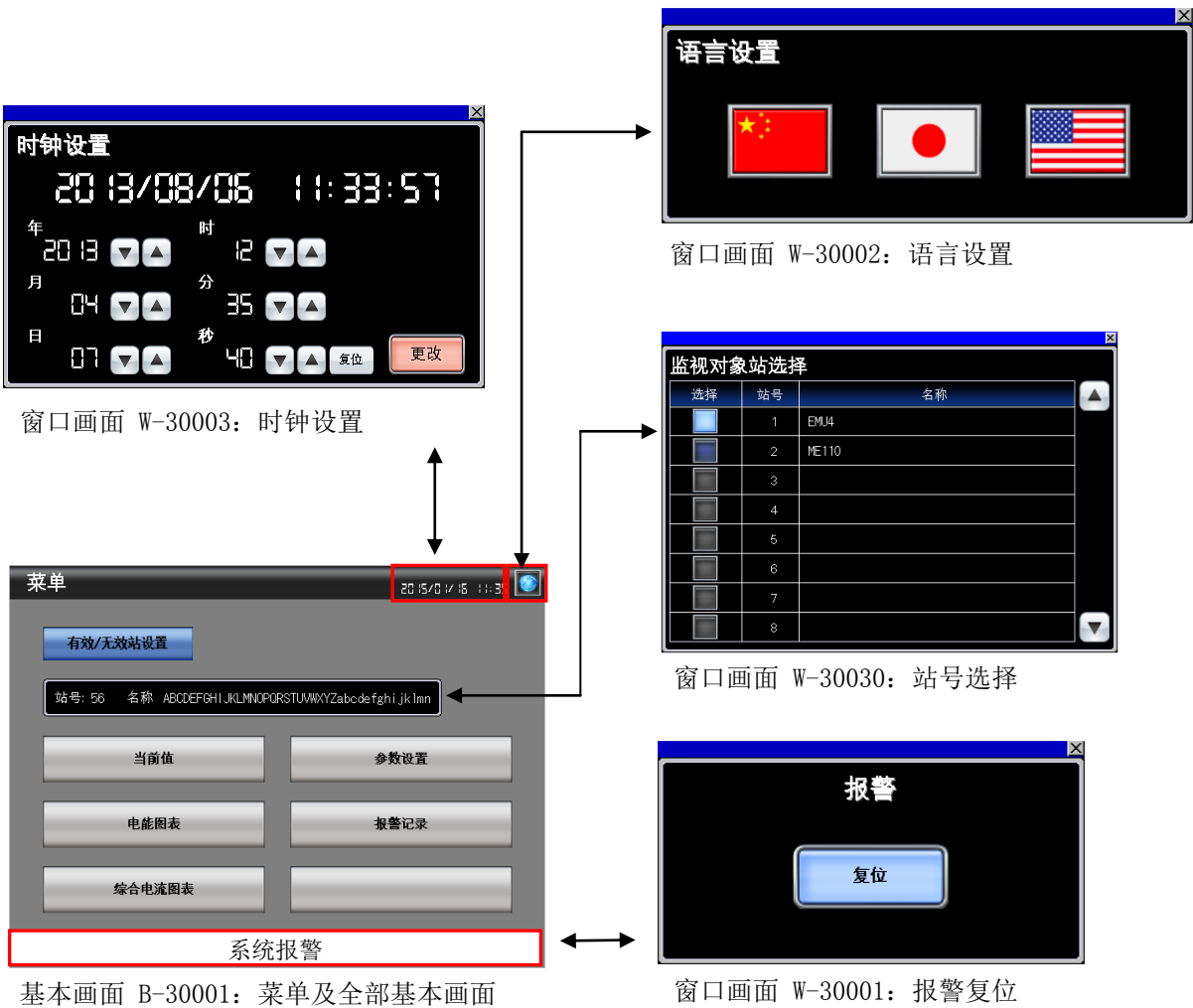
5.1 显示语言

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 254、255 的列号 1~3 中。将列号写入语言切换软件元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

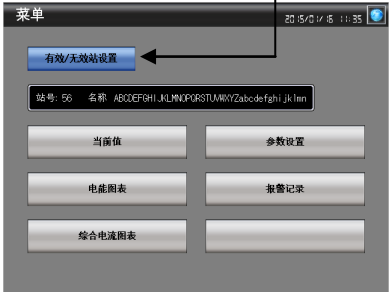
5.2.1 画面一览表/切换(公共)



5.2.2 画面一览表/切换(个别)



窗口画面 W-30031:
有效/无效站设置



基本画面 B-30001: 菜单



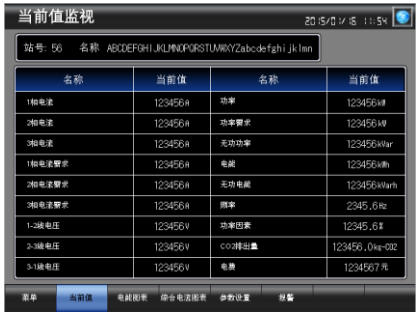
窗口画面 W-30015:
相线系统



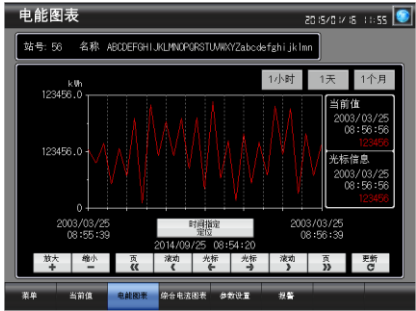
窗口画面 W-30016:
传感器类型



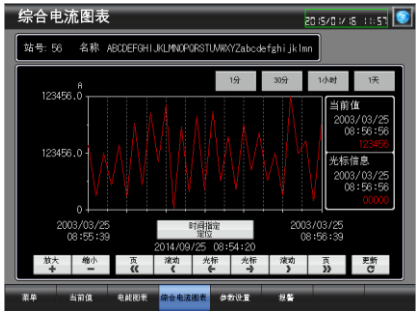
窗口画面 W-30017~30020:
各种 乘比率



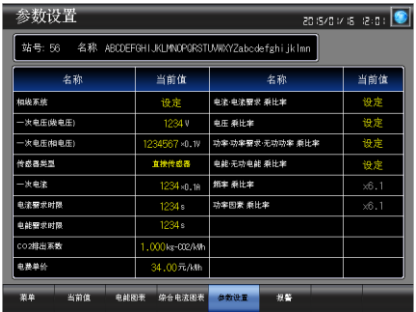
基本画面 B-30002: 当前值监视



基本画面 B-30009: 电能图表



基本画面 B-30010: 综合电流图表



基本画面 B-30020: 参数设置



基本画面 B-30011: 报警记录

5.3 画面说明

5.3.1 菜单(B-30001)



概要

菜单画面。

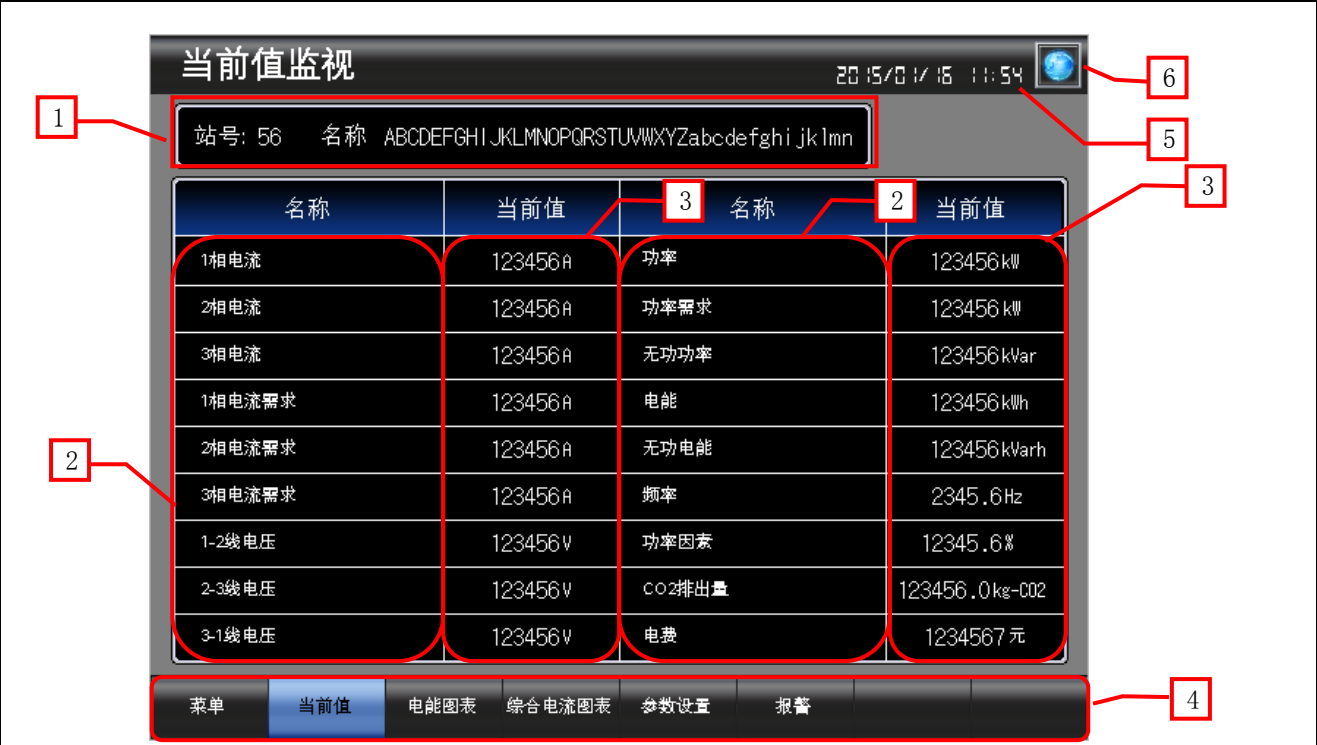
详细

- 1. 显示用于设置测定终端的站号有效/无效、名称、机种的窗口。
- 2. 显示当前监视的测定终端的站号和名称。触摸即显示用于切换监视站的窗口。
- 3. 切换到当前值显示画面。
- 4. 切换到电能图表画面。
- 5. 切换到综合电流图表画面。
- 6. 切换到参数设置画面。
- 7. 切换到报警记录画面。
- 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

- 在 GOT 起动时，通过工程脚本从扩展配方文件中读出 1 站的信息，设置为监视对象站。关于脚本的详细情况，请参照“5.7 脚本一览表”。
- 发生系统报警时，在画面下方将显示报警信息。选择报警信息，则显示报警复位窗口。

5.3.2 当前值监视(B-30002)



概要

监视 EcoMonitor 的当前值。当前值的小数点位置因乘比率而异。

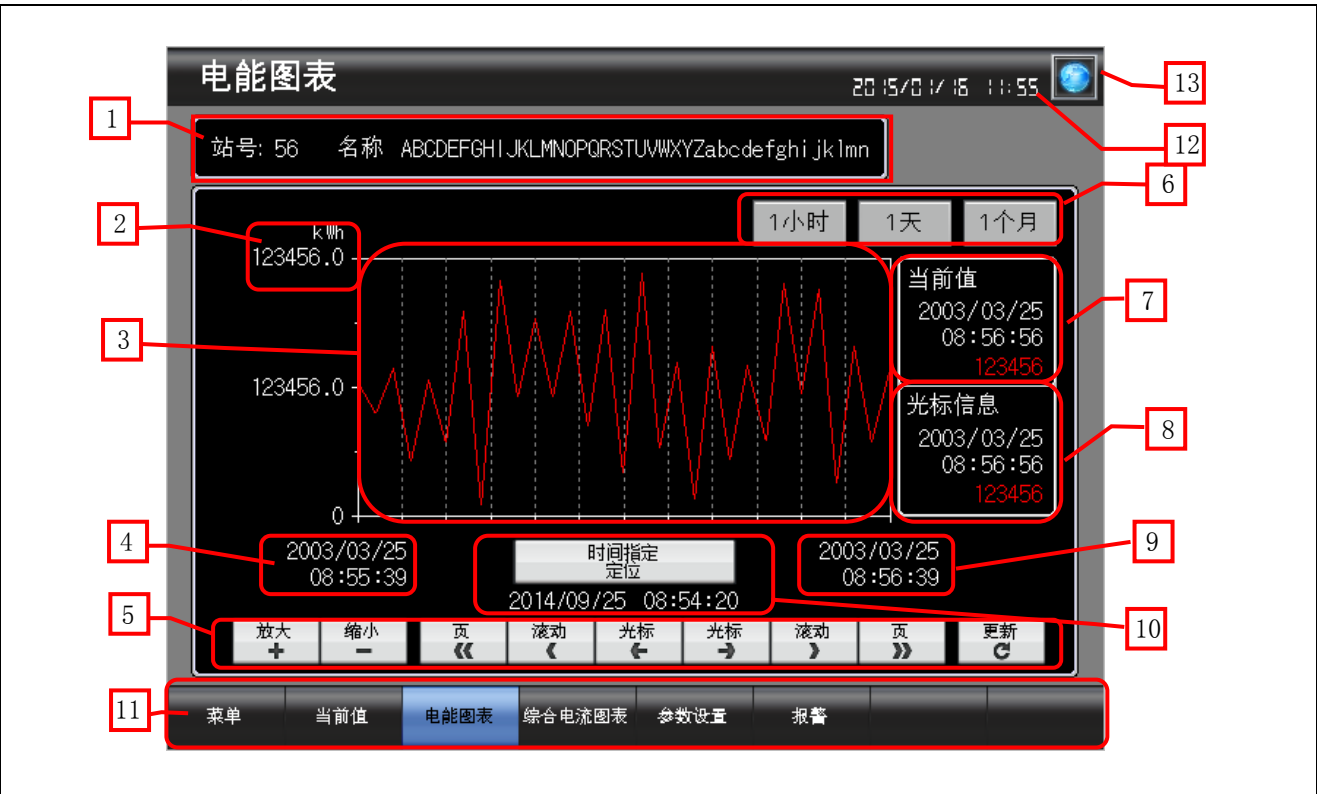
详细

1. 显示当前监视的测定终端的站号和名称。触摸即显示用于切换监视站的窗口。
2. 显示当前监视的测定值的名称。
3. 显示当前监视的测定终端的当前值。
4. 切换至各画面。蓝色开关为当前显示的画面，所以显示中的画面不被切换。
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

- 乘比率类型如下所示。
机种 EMU4: $\times 0.001$ 、 $\times 0.01$ 、 $\times 0.1$ 、 $\times 1$
机种 ME110: $\times 0.01$ 、 $\times 0.1$ 、 $\times 1$ 、 $\times 10$
- 在更改相线系统、一次电压、一次电流时，机种 EMU4 的乘比率将会自动更改。机种 ME110 的乘比率由用户设置。
- 根据以下计算，算出 CO2 排出量。
 $\text{CO2 排出量 (kg-CO2)} = \text{电能 (kWh)} \times \text{CO2 排出系数 (kg-CO2/kWh)}$
- CO2 排出量的显示不同于测定终端侧的 CO2 排出量 (CO2 量换算值)。
- 根据以下计算，算出电费。
 $\text{电费 (日元 or \$ or 元)} = \text{电能 (kWh)} \times \text{电费单价 (日元 or \$ or 元 /kWh)}$
- 发生系统报警时，在画面下方将显示报警信息。选择报警信息，则显示报警复位窗口。

5.3.3 电能图表(B-30009)



概要

显示电能的当前值，通过记录趋势图表显示由日志功能收集的过去电能。

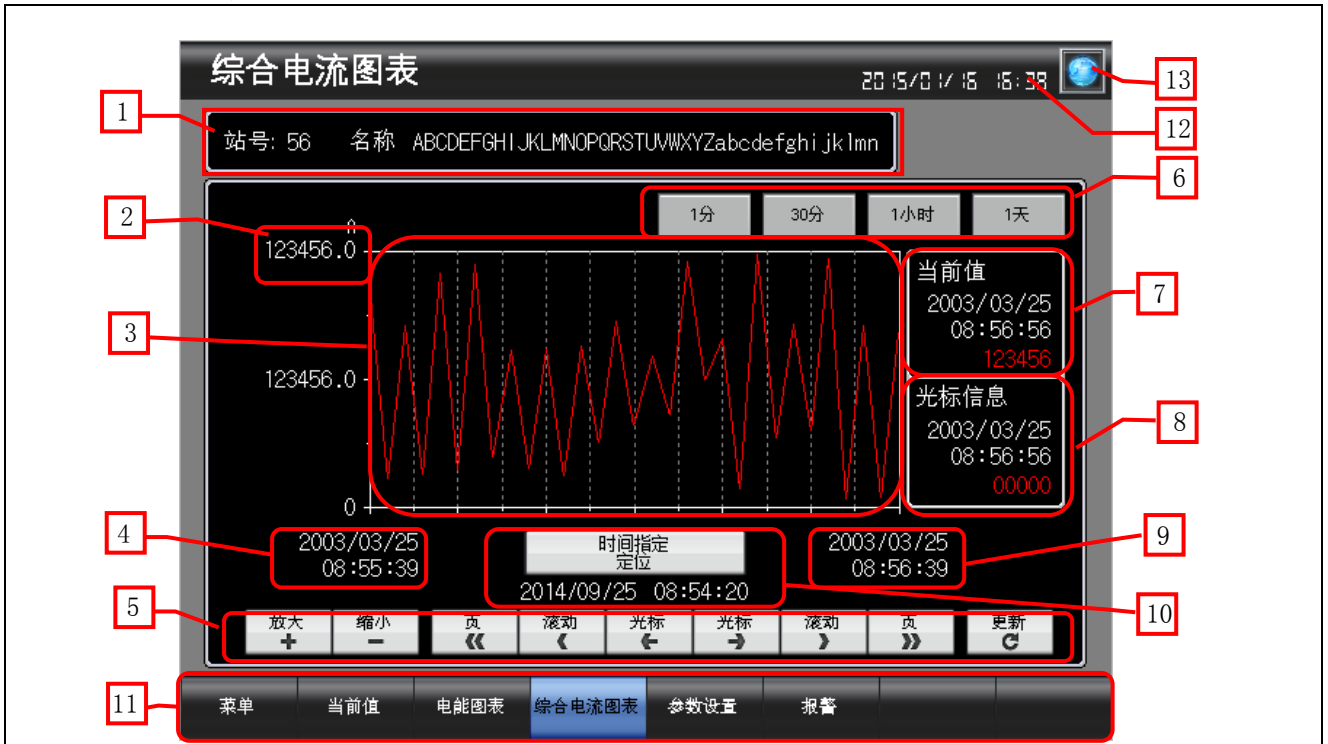
详细

1. 显示当前监视的测定终端的站号和名称。触摸即显示用于切换监视站的窗口。
2. 显示记录趋势图表的上限值。按下数值即可输入上限值。输入的上限值通过扩展配方功能保存，并保持其数值。切换站号时，将切换到与站号对应的上限值。
3. 用记录趋势图表显示电能。按下趋势图表，可显示光标。横轴的数据数为 31 点。
4. 显示记录趋势图表的显示结束位置时间。
5. 操作记录趋势图表。
6. 切换用记录趋势图表显示的日志数据。
 - “1 小时”：每 00 分收集到的每 1 小时的日志数据。
 - “1 天”：每 00 时 00 分收集到的每 1 天的日志数据。
 - “1 个月”：每 01 日 00 时 00 分收集到的每 1 个月的日志数据。
7. 显示电能的当前值。小数点位置因乘比率而异。
8. 显示光标位置的日期时间和电能。电能的小数点位置因乘比率而异。
9. 显示记录趋势图表的显示开始位置时间。
10. 将光标移动到任意指定的时间。
11. 切换至各画面。蓝色开关为当前显示的画面，所以显示中的画面不被切换。
12. 显示当前日期和时间。触摸即显示时钟设置窗口。
13. 显示语言设置窗口。

备注

- 通过软元件数据传送功能将电能从测定终端传送到 GOT 的内部软元件。使用脚本将传送的值转换为实数，用日志功能收集转换后的数据。因此，如果转换为实数后的值超过 7 位，可能会发生误差。脚本的详细内容请参照“5.7 脚本一览表”。
- 发生系统报警时，在画面下方将显示报警信息。选择报警信息，则显示报警复位窗口。

5.3.4 综合电流图表(B-30010)



概要

显示综合电流的当前值，通过记录趋势图表显示由日志功能收集的过去的综合电流。

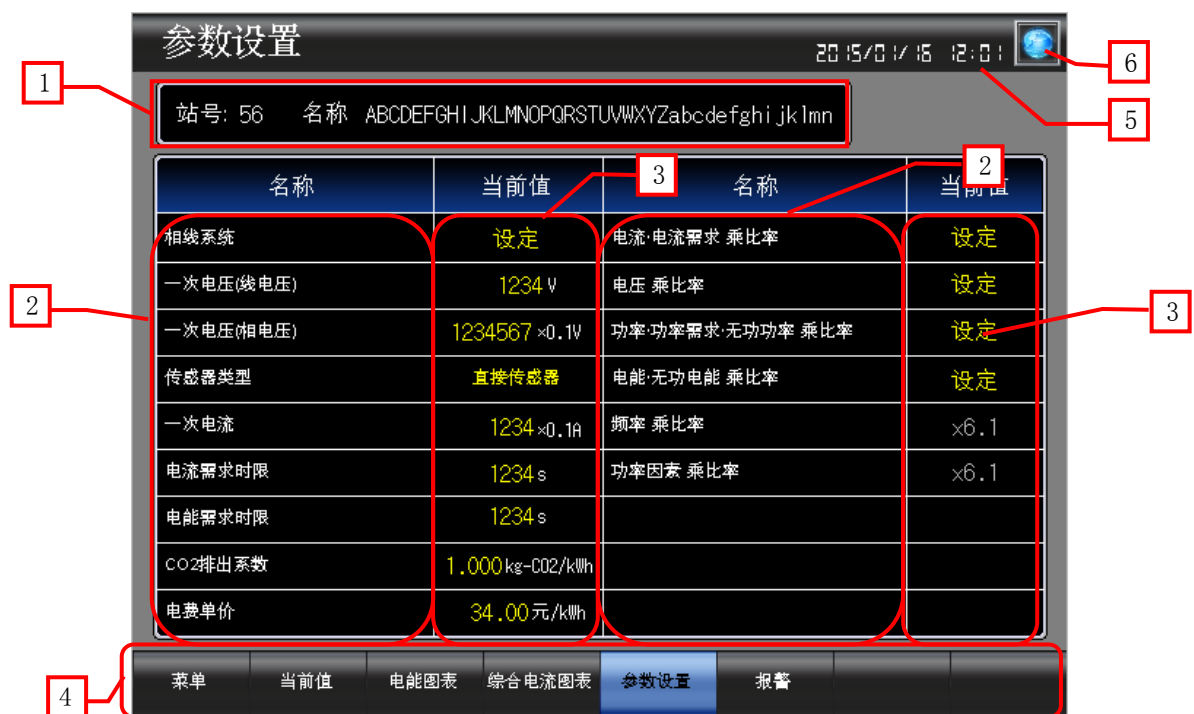
详细

1. 显示当前监视的测定终端的站号和名称。触摸即显示用于切换监视站的窗口。
2. 显示记录趋势图表的上限值。按下数值即可输入上限值。输入的上限值通过扩展配方功能保存，并保持其数值。切换站号时，将切换到与站号对应的上限值。
3. 用记录趋势图表显示综合电流。按下趋势图表，可显示光标。横轴的数据数为 31 点。
4. 显示记录趋势图表的显示结束位置时间。
5. 操作记录趋势图表。
6. 切换用记录趋势图表显示的登录数据。
 - “1 分”：每 00 秒收集到的每 1 分钟的日志数据。
 - “30 分”：每 00 分和每 30 分收集到的每 30 分钟的日志数据。
 - “1 小时”：每 00 分收集到的每 1 小时的日志数据。
 - “1 天”：每 00 时 00 分收集到的每 1 天的日志数据。
7. 显示综合电流的当前值。小数点位置因乘比率而异。
8. 显示光标位置的日期时间和综合电流。综合电流的小数点位置因乘比率而异。
9. 显示记录趋势图表的显示开始位置时间。
10. 将光标移动到任意指定的时间。
11. 切换至各画面。蓝色开关为当前显示的画面，所以显示中的画面不被切换。
12. 显示当前日期和时间。触摸即显示时钟设置窗口。
13. 显示语言设置窗口。

备注

- 通过软元件数据传送功能将电能从测定终端传送到 GOT 的内部软元件。使用脚本将传送的值转换为实数，用日志功能收集转换后的数据。因此，如果转换为实数后的值超过 7 位，可能会发生误差。脚本的详细内容请参照“5.7 脚本一览表”。
- 机种 ME110 中，如果设置为 3P4W 以外的相线系统，则综合电流图表为 0 显示。
- 发生系统报警时，在画面下方将显示报警信息。选择报警信息，则显示报警复位窗口。

5.3.5 参数设置(B-30020)



概要

执行 EcoMonitor 的参数设置。

详细

1. 显示当前监视的测定终端的站号和名称。触摸即显示用于切换监视站的窗口。
2. 显示设置值的名称。
3. 按下设置值即显示用于设置各参数的窗口、或者显示用于输入数值的按键窗口。
4. 切换至各画面。蓝色开关为当前显示的画面，所以显示中的画面不被切换。
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

- 发生系统报警时，在画面下方将显示报警信息。选择报警信息，则显示报警复位窗口。

5.3.6 报警记录(B-30011)



概要

显示测定终端的报警和异常状态。

详细

1. 以报警显示(用户)显示报警。
2. 将报警显示(用户)进行操作。
删除: 删除所选择的已恢复报警。
全删除: 删除所有已恢复报警。
3. 将报警显示(用户)进行滚动操作。
▲▼: 上下滚动页面。
▼▲: 上下移动光标。
4. 将报警显示(用户)中显示的报警, 进行监视站和全部站点的切换。
5. 切换至各画面。蓝色开关为当前显示的画面, 所以显示中的画面不被切换。
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

- 发生系统报警时, 在画面下方将显示报警信息。选择报警信息, 则显示报警复位窗口。

5.3.7 报警复位(W-30001)



概要

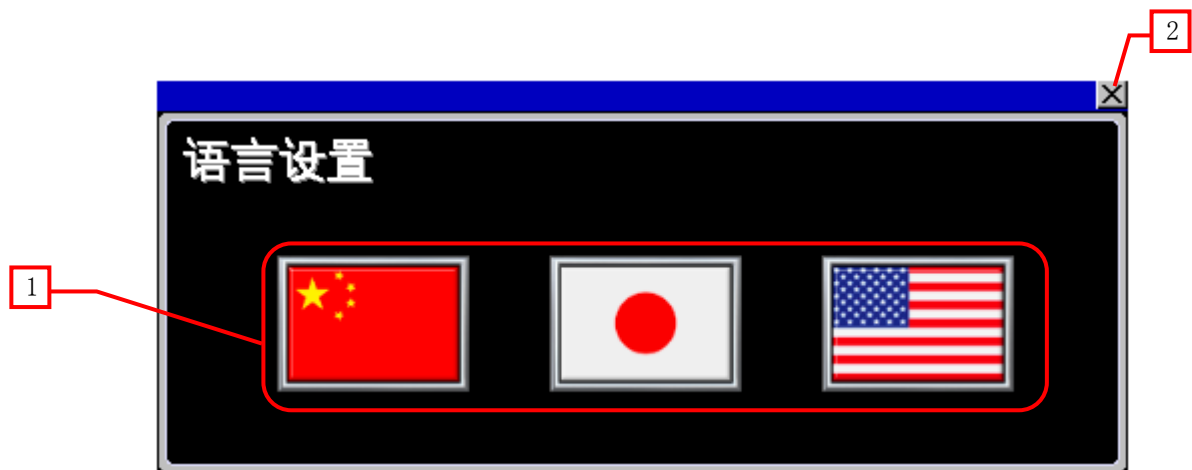
复位系统报警。

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

5.3.8 语言设置(W-30002)



概要

选择 GOT 的显示语言。

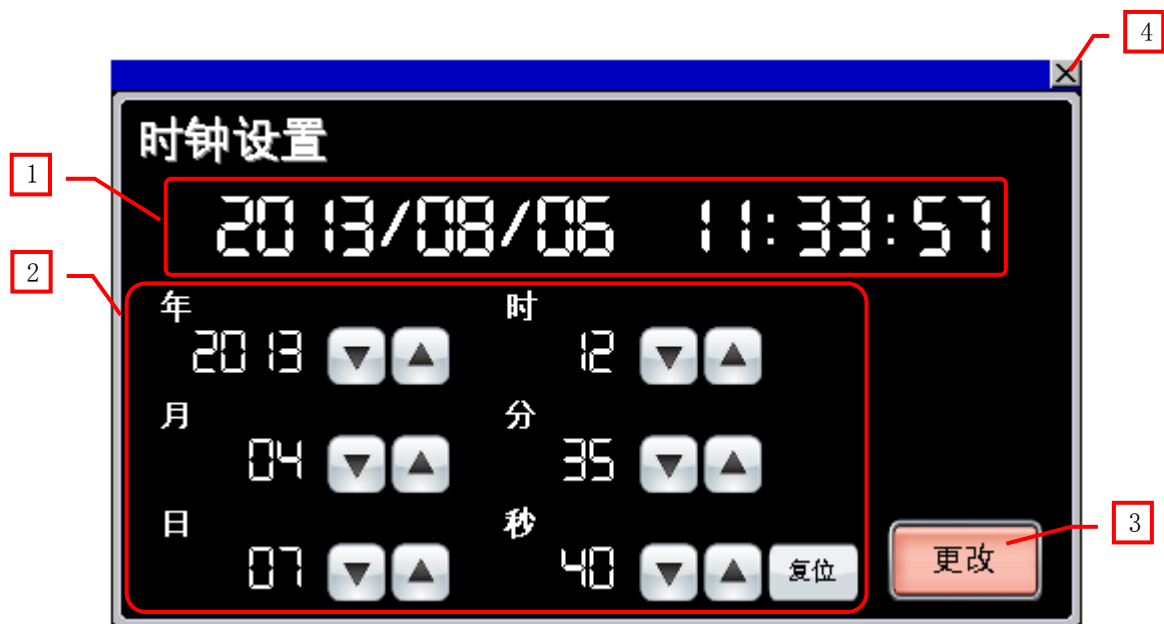
详细

1. 切换语言，关闭窗口画面。
2. 关闭窗口画面。

备注

- 画面显示语言与系统语言同步切换。

5.3.9 时钟设置(W-30003)



概要

更改 GOT 的时钟数据。

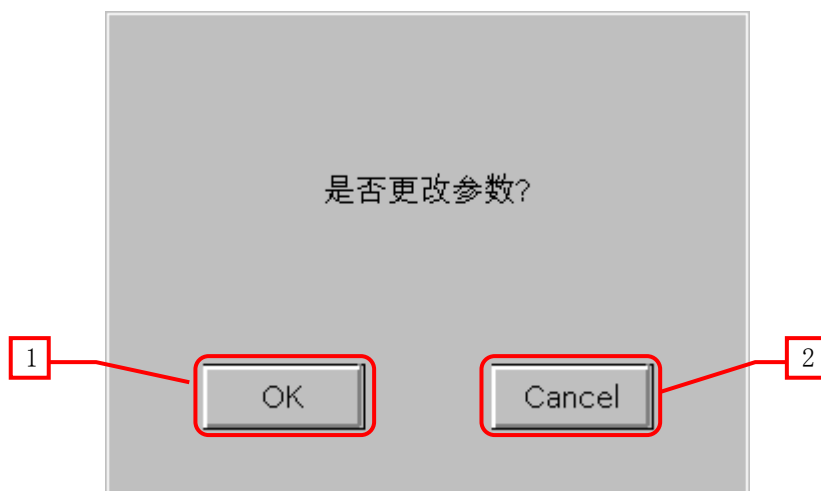
详细

1. 显示当前日期和时间。
2. 通过 设置要更改日期和时间。长按 开关将连续进行增减。复位开关可复位秒。
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显示窗口画面时的日期和时间。
- 在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。关于脚本的详细内容，请参照“5.7 脚本一览表”。

5.3.10 输入确认对话框(W-30014)



概要

在确定各参数设置用窗口画面中所设置的值时显示的对话框。

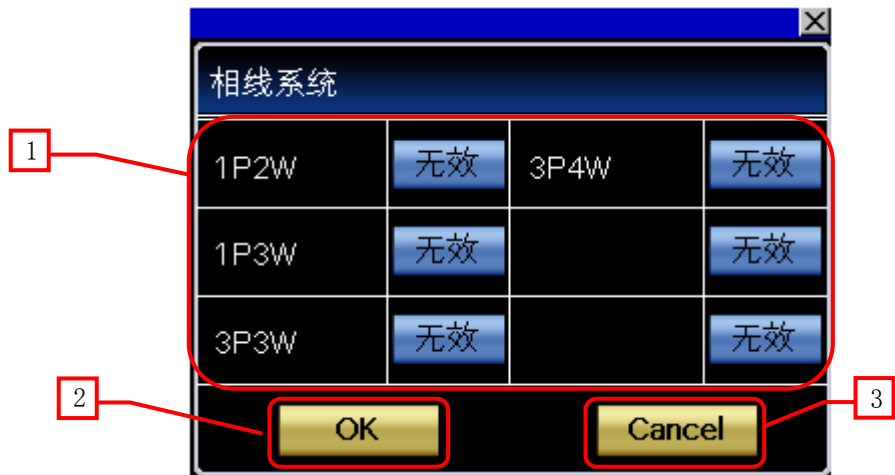
详细

1. 设置输入的值，关闭对话框。
2. 对输入结果不进行确定，关闭对话框。

备注

- 以下参数的值在 GOT 侧即使停电也可保持数据，保存在配方文件中。
机种 EMU4、机种 ME110 公共：CO2 排出系数、电费单价
仅限机种 ME110：各种乘比率(除频率、功率因素以外)

5.3.11 相线系统(W-30015)



概要

设置相线系统。相线系统的设置内容因机种而异。

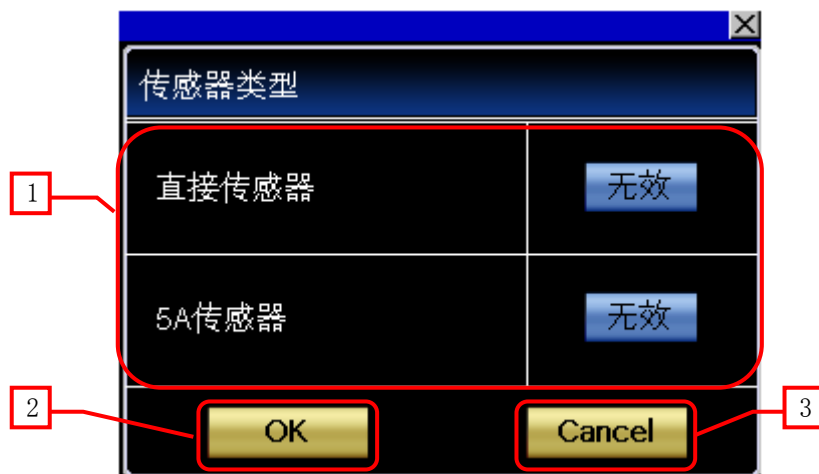
详细

1. 选择要设置的相线系统。各机种可设置的相线系统如下所示。
机种 EMU4: 1P2W、1P3W、3P3W、3P4W
机种 ME110: 1P2W、1P3W(RNT)、3P3W、3P4W、1P3W(RNS)、3P3W_3CT
2. 显示确认对话框窗口，按下 OK，确定输入内容。
3. 放弃设置的值，关闭窗口。

备注

- 机种 ME110 时，相线系统的更改处理需要 2~3 秒。此期间如果更改各种参数的设置，可能会发生通信错误，敬请注意。

5.3.12 传感器类型 (W-30016)



概要

设置机种 EMU4 的传感器类型。

详细

1. 选择要设置的传感器类型。
2. 显示确认对话框窗口，按下 OK，确定输入内容。
3. 放弃设置的值，关闭窗口。

备注

5.3.13 电流・电流需求 乘比率(W-30017)



概要

设置机种 ME110 的电流・电流需求 乘比率。

详细

1. 选择要设置的乘比率。
2. 显示确认对话框窗口，按下 OK，确定输入内容。
3. 放弃设置的值，关闭窗口。

备注

5.3.14 电压 乘比率(W-30018)



概要

设置机种 ME110 的电压 乘比率。

详细

1. 选择要设置的乘比率。
2. 显示确认对话框窗口，按下 OK，确定输入内容。
3. 放弃设置的值，关闭窗口。

备注

5.3.15 功率・功率需求・无功功率 乘比率(W-30019)



概要

设置机种 ME110 的功率・功率需求・无功功率 乘比率。

详细

1. 选择要设置的乘比率。
2. 显示确认对话框窗口，按下 OK，确定输入内容。
3. 放弃设置的值，关闭窗口。

备注

5.3.16 电能・无功电能 乘比率(W-30020)



概要

设置机种 ME110 的电能・无功电能 乘比率。

详细

1. 选择要设置的乘比率。
2. 显示确认对话框窗口，按下 OK，确定输入内容。
3. 放弃设置的值，关闭窗口。

备注

5.3.17 有效/无效站设置(W-30031)



概要

切换测定终端站号的有效/无效，设置名称和机种。

详细

1. 切换测定终端的站号有效/无效.
2. 设置所指定站号的测定终端的名称.
3. 选择所指定站号的测定终端的机种。可选择机种 EMU4 或机种 ME110。
4. 通过 ▼ ▲ 开关切换显示站号。
5. 关闭窗口画面。

备注

5. 3. 18 站号选择(W-30030)



概要

切换监视对象站。

详细

1. 切换监视对象站。
2. 通过有效站/无效站设置，显示所指定的测定终端的名称。
3. 通过▼▲ 开关切换显示站号。
4. 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使用的软元件，有些同时也在脚本等的公共设置中被使用。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	机种 EMU4	机种 ME110
位	1-1~1-31: 400525.b2	电流需求上下限报警	○	○
	1-1~1-31: 400525.b3	功率需求上下限报警	○	○
	1-1~1-31: 400525.b8	电压上下限报警	○	○
	1-1~1-31: 400525.b9	电流上下限报警	—	○
	1-1~1-31: 400525.b10	功率上下限报警	—	○
	1-1~1-31: 400525.b11	无功功率上下限报警	—	○
	1-1~1-31: 400525.b12	频率上下限报警	—	○
	1-1~1-31: 400525.b13	功率因素上下限报警	○	○
	1-1~1-31: 400525.b14	高频波电压上限报警	—	○
	1-1~1-31: 400525.b15	高频波电流上限报警	—	○
字	400513	相线系统	○	○
	400514	一次电压(线电压)	○	○
	400516	一次电压(相电压)	○	○
	400520	一次电流	○	○
	400522	功率需求时限	○	○
	400523	电流需求时限	○	○
	400532	传感器类型	○	—
	400755	电流・电流需求 乘比率	○	—
	400756	电压 乘比率	○	—
	400757	功率・功率需求・无功功率 乘比率	○	—
	400758	电能・无功电能 乘比率	○	—
	400769	1 相电流	○	○
	400770	2 相电流	○	○
	400771	3 相电流	○	○
	400773	电流当前值(平均)	○	○
	400774	1 相电流需求	○	○
	400775	2 相电流需求	○	○
	400776	3 相电流需求	○	○
	400779	1-2 线电压	○	○
	400780	2-3 线电压	○	○
	400781	3-1 线电压	○	○
	400790	功率因素	○	○
	400791	频率	○	○
	400795	功率	○	○
	400799	功率需求	○	○
	400803	无功功率	○	○
	401305	电能	○	○

类型	软元件编号	用途	机种 EMU4	机种 ME110
字	401309	无功电能	○	○

○：有软元件、-：无软元件

5.4.2 GOT 内部软元件

类型	软元件编号	用途
位	GB40	脚本触发(始终 ON)
	GB61000	日志 ID: 30001 触发软元件
	GB61001	日志 ID: 30002 触发软元件
	GB61002	日志 ID: 30003 触发软元件
	GB61003	日志 ID: 30004 触发软元件
	GB61004	日志 ID: 30005 触发软元件
	GB61005	日志 ID: 30006 触发软元件
	GB61006	日志 ID: 30007 触发软元件
	GB61007	脚本 No. 30005 触发软元件
	GB61008	脚本 No. 30006 触发软元件
	GB61009	脚本 No. 30014 触发软元件
	GB61010	脚本 No. 30017、30023 触发软元件
	GB61011	脚本 No. 30004、30024 触发软元件
	GB61012	脚本 No. 30020 触发软元件
	GB61013	脚本 No. 30021 触发软元件
	GB61014	脚本 No. 30025 触发软元件
	GB61015	脚本 No. 30030 配方读出标志
	GB61016	一次电压(线电压、相电压)显示/隐藏标志
	GB61017	日志用软元件数据传送实施时机控制标志
	GB61018	脚本 No. 30018 触发软元件
	GB61019	脚本 No. 30034 触发软元件
	GB61020	脚本 No. 30035 触发软元件
	GB61021	有效/无效站状态标志(偏置指定)
	GB61022	有效/无效站状态标志(偏置指定)
	GB61023	有效/无效站状态标志(偏置指定)
	GB61024	有效/无效站状态标志(偏置指定)
	GB61025	有效/无效站状态标志(偏置指定)
	GB61026	有效/无效站状态标志(偏置指定)
	GB61027	有效/无效站状态标志(偏置指定)
	GB61028	有效/无效站状态标志(偏置指定)
	GB61052	监视对象站选择开关 动作条件
	GB61053	监视对象站选择开关 动作条件
	GB61054	监视对象站选择开关 动作条件
	GB61055	监视对象站选择开关 动作条件
	GB61056	监视对象站选择开关 动作条件
	GB61057	监视对象站选择开关 动作条件
	GB61058	监视对象站选择开关 动作条件
	GB61059	监视对象站选择开关 动作条件
	GB61060	站号选择指示灯(偏置指定)
	GB61061	站号选择指示灯(偏置指定)
	GB61062	站号选择指示灯(偏置指定)
	GB61063	站号选择指示灯(偏置指定)
	GB61064	站号选择指示灯(偏置指定)
	GB61065	站号选择指示灯(偏置指定)
	GB61066	站号选择指示灯(偏置指定)
	GB61067	站号选择指示灯(偏置指定)
	GB61091	脚本 No. 30036 触发软元件
	GB61092	站号 32 指示灯显示/隐藏标志

类型	软元件编号	用途
位	GB61093	脚本 No. 30039 触发软元件
	GD60031. b13	GOT 错误复位信号
	GD62557. b4	脚本 No. 30002 触发软元件
	GD62557. b5	脚本 No. 30015 触发软元件
	GD62560. b0	配方 1 写入触发软元件
	GD62560. b1	配方 1 读取触发软元件
	GD62562. b1	软元件数据传送 ID: 193、194 完成标志
	GD62562. b2	软元件数据传送 ID: 195、196 完成标志
	GD62562. b3	软元件数据传送 ID: 197、198 完成标志
	GD62562. b4	软元件数据传送 ID: 199、200 完成标志
	GD62562. b5	软元件数据传送 ID: 201、202 完成标志
	GD62562. b6	软元件数据传送 ID: 203、204 完成标志
	GD62562. b7	软元件数据传送 ID: 205、206 完成标志
	GD62562. b8	软元件数据传送 ID: 207、208 完成标志
	GD62562. b9	软元件数据传送 ID: 209、210 完成标志
	GD62562. b10	软元件数据传送 ID: 211、212 完成标志
	GD62562. b11	软元件数据传送 ID: 213、214 完成标志
	GD62562. b12	软元件数据传送 ID: 215、216 完成标志
	GD62562. b13	软元件数据传送 ID: 217、218 完成标志
	GD62562. b14	软元件数据传送 ID: 219、220 完成标志
	GD62562. b15	软元件数据传送 ID: 221、222 完成标志
	GD62563. b0	软元件数据传送 ID: 223、224 完成标志
	GD62563. b1	软元件数据传送 ID: 225、226 完成标志
	GD62563. b2	软元件数据传送 ID: 227、228 完成标志
	GD62563. b3	软元件数据传送 ID: 229、230 完成标志
	GD62563. b4	软元件数据传送 ID: 231、232 完成标志
	GD62563. b5	软元件数据传送 ID: 233、234 完成标志
	GD62563. b6	软元件数据传送 ID: 235、236 完成标志
	GD62563. b7	软元件数据传送 ID: 237、238 完成标志
	GD62563. b8	软元件数据传送 ID: 239、240 完成标志
	GD62563. b9	软元件数据传送 ID: 241、242 完成标志
	GD62563. b10	软元件数据传送 ID: 243、244 完成标志
	GD62563. b11	软元件数据传送 ID: 245、246 完成标志
	GD62563. b12	软元件数据传送 ID: 247、248 完成标志
	GD62563. b13	软元件数据传送 ID: 249、250 完成标志
	GD62563. b14	软元件数据传送 ID: 251、252 完成标志
	GD62563. b15	软元件数据传送 ID: 253、254 完成标志
	GD62640. b0	软元件数据传送 ID: 193 触发软元件
	GD62642. b0	软元件数据传送 ID: 194 触发软元件
	GD62644. b0	软元件数据传送 ID: 195 触发软元件
	GD62646. b0	软元件数据传送 ID: 196 触发软元件
	GD62648. b0	软元件数据传送 ID: 197 触发软元件
	GD62650. b0	软元件数据传送 ID: 198 触发软元件
	GD62652. b0	软元件数据传送 ID: 199 触发软元件
	GD62654. b0	软元件数据传送 ID: 200 触发软元件
	GD62656. b0	软元件数据传送 ID: 201 触发软元件
	GD62658. b0	软元件数据传送 ID: 202 触发软元件
	GD62660. b0	软元件数据传送 ID: 203 触发软元件
	GD62662. b0	软元件数据传送 ID: 204 触发软元件
	GD62664. b0	软元件数据传送 ID: 205 触发软元件
	GD62666. b0	软元件数据传送 ID: 206 触发软元件
	GD62668. b0	软元件数据传送 ID: 207 触发软元件
	GD62670. b0	软元件数据传送 ID: 208 触发软元件

类型	软元件编号	用途
位	GD62672. b0	软元件数据传送 ID: 209 触发软元件
	GD62674. b0	软元件数据传送 ID: 210 触发软元件
	GD62676. b0	软元件数据传送 ID: 211 触发软元件
	GD62678. b0	软元件数据传送 ID: 212 触发软元件
	GD62680. b0	软元件数据传送 ID: 213 触发软元件
	GD62682. b0	软元件数据传送 ID: 214 触发软元件
	GD62684. b0	软元件数据传送 ID: 215 触发软元件
	GD62686. b0	软元件数据传送 ID: 216 触发软元件
	GD62688. b0	软元件数据传送 ID: 217 触发软元件
	GD62690. b0	软元件数据传送 ID: 218 触发软元件
	GD62692. b0	软元件数据传送 ID: 219 触发软元件
	GD62694. b0	软元件数据传送 ID: 220 触发软元件
	GD62696. b0	软元件数据传送 ID: 221 触发软元件
	GD62698. b0	软元件数据传送 ID: 222 触发软元件
	GD62700. b0	软元件数据传送 ID: 223 触发软元件
	GD62702. b0	软元件数据传送 ID: 224 触发软元件
	GD62704. b0	软元件数据传送 ID: 225 触发软元件
	GD62706. b0	软元件数据传送 ID: 226 触发软元件
	GD62708. b0	软元件数据传送 ID: 227 触发软元件
	GD62710. b0	软元件数据传送 ID: 228 触发软元件
	GD62712. b0	软元件数据传送 ID: 229 触发软元件
	GD62714. b0	软元件数据传送 ID: 230 触发软元件
	GD62716. b0	软元件数据传送 ID: 231 触发软元件
	GD62718. b0	软元件数据传送 ID: 232 触发软元件
	GD62720. b0	软元件数据传送 ID: 233 触发软元件
	GD62722. b0	软元件数据传送 ID: 234 触发软元件
	GD62724. b0	软元件数据传送 ID: 235 触发软元件
	GD62726. b0	软元件数据传送 ID: 236 触发软元件
	GD62728. b0	软元件数据传送 ID: 237 触发软元件
	GD62730. b0	软元件数据传送 ID: 238 触发软元件
	GD62732. b0	软元件数据传送 ID: 239 触发软元件
	GD62734. b0	软元件数据传送 ID: 240 触发软元件
	GD62736. b0	软元件数据传送 ID: 241 触发软元件
	GD62738. b0	软元件数据传送 ID: 242 触发软元件
	GD62740. b0	软元件数据传送 ID: 243 触发软元件
	GD62742. b0	软元件数据传送 ID: 244 触发软元件
	GD62744. b0	软元件数据传送 ID: 245 触发软元件
	GD62746. b0	软元件数据传送 ID: 246 触发软元件
	GD62748. b0	软元件数据传送 ID: 247 触发软元件
	GD62750. b0	软元件数据传送 ID: 248 触发软元件
	GD62752. b0	软元件数据传送 ID: 249 触发软元件
	GD62754. b0	软元件数据传送 ID: 250 触发软元件
	GD62756. b0	软元件数据传送 ID: 251 触发软元件
	GD62758. b0	软元件数据传送 ID: 252 触发软元件
	GD62760. b0	软元件数据传送 ID: 253 触发软元件
	GD62762. b0	软元件数据传送 ID: 254 触发软元件
	GS512. b0	时间更改信号
字	GD10	站号的间接指定
	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
	GD60018	对话框窗口
	GD60021	语言切换

类型	软元件编号	用途
字	GD60022	系统语言切换
	GD60031、GD60041	系统信息
	GD61000~GD61039	站号 1 各种参数 (参照站号 1 名称~传感器类型)
	GD61040~GD61079	站号 2 各种参数 (参照站号 1 名称~传感器类型)
	GD61080~GD61119	站号 3 各种参数 (参照站号 1 名称~传感器类型)
	GD61120~GD61159	站号 4 各种参数 (参照站号 1 名称~传感器类型)
	GD61160~GD61199	站号 5 各种参数 (参照站号 1 名称~传感器类型)
	GD61200~GD61239	站号 6 各种参数 (参照站号 1 名称~传感器类型)
	GD61240~GD61279	站号 7 各种参数 (参照站号 1 名称~传感器类型)
	GD61280~GD61319	站号 8 各种参数 (参照站号 1 名称~传感器类型)
	GD61320~GD61359	站号 9 各种参数 (参照站号 1 名称~传感器类型)
	GD61360~GD61399	站号 10 各种参数 (参照站号 1 名称~传感器类型)
	GD61400~GD61439	站号 11 各种参数 (参照站号 1 名称~传感器类型)
	GD61440~GD61479	站号 12 各种参数 (参照站号 1 名称~传感器类型)
	GD61480~GD61519	站号 13 各种参数 (参照站号 1 名称~传感器类型)
	GD61520~GD61559	站号 14 各种参数 (参照站号 1 名称~传感器类型)
	GD61560~GD61599	站号 15 各种参数 (参照站号 1 名称~传感器类型)
	GD61600~GD61639	站号 16 各种参数 (参照站号 1 名称~传感器类型)
	GD61640~GD61679	站号 17 各种参数 (参照站号 1 名称~传感器类型)
	GD61680~GD61719	站号 18 各种参数 (参照站号 1 名称~传感器类型)
	GD61720~GD61759	站号 19 各种参数 (参照站号 1 名称~传感器类型)
	GD61760~GD61799	站号 20 各种参数 (参照站号 1 名称~传感器类型)
	GD61800~GD61839	站号 21 各种参数 (参照站号 1 名称~传感器类型)
	GD61840~GD61879	站号 22 各种参数 (参照站号 1 名称~传感器类型)
	GD61880~GD61919	站号 23 各种参数 (参照站号 1 名称~传感器类型)
	GD61920~GD61959	站号 24 各种参数 (参照站号 1 名称~传感器类型)
	GD61960~GD61999	站号 25 各种参数 (参照站号 1 名称~传感器类型)
	GD62000~GD62039	站号 26 各种参数 (参照站号 1 名称~传感器类型)
	GD62040~GD62079	站号 27 各种参数 (参照站号 1 名称~传感器类型)
	GD62080~GD62119	站号 28 各种参数 (参照站号 1 名称~传感器类型)
	GD62120~GD62159	站号 29 各种参数 (参照站号 1 名称~传感器类型)
	GD62160~GD62199	站号 30 各种参数 (参照站号 1 名称~传感器类型)
	GD62200~GD62239	站号 31 各种参数 (参照站号 1 名称~传感器类型)
	GD62240~GD62301	站号 1~31 电能日志软元件
	GD62302~GD62362	站号 1~31 综合电流日志软元件
	GD62364~GD62425	站号 1~31 电能 软元件数据传送目标
	GD62426~GD62456	站号 1~31 综合电流 软元件数据传送目标
	GD62457~GD62487	站号 1~31 综合电流 乘比率 软元件数据传送目标
	GD62488~GD62518	站号 1~31 电能 乘比率 软元件数据传送目标
	GD62552	参数偏置软元件
	GD62553	现在机种保存软元件
	GD62554	配方公共设置 外部控制软元件
	GD62555	配方公共设置 配方号保存软元件
	GD62556	配方公共设置 记录号保存软元件
	GD62557	配方公共设置 外部通知软元件
	GD62558	配方公共设置 配方号通知软元件
	GD62559	配方公共设置 记录号通知软元件
	GD62560	配方 No. 30001 写入/读出触发
	GD62561	配方 No. 30001 记录号
	GD62562	软元件数据传送完成检测标志
	GD62563	软元件数据传送完成检测标志
	GD62564	电能图表 日志 ID 软元件
	GD62565	电能图表 图表偏置软元件

类型	软元件编号	用途
字	GD62566	电能图表 上限值软元件
	GD62567	电能图表 上限值软元件
	GD62568	电能图表 光标位置时间
	GD62569	电能图表 光标位置时间
	GD62570	电能图表 光标位置时间
	GD62571	电能图表 光标位置时间
	GD62572	电能图表 显示开始位置时间
	GD62573	电能图表 显示开始位置时间
	GD62574	电能图表 显示开始位置时间
	GD62575	电能图表 显示开始位置时间
	GD62576	电能图表 显示结束位置时间
	GD62577	电能图表 显示结束位置时间
	GD62578	电能图表 显示结束位置时间
	GD62579	电能图表 显示结束位置时间
	GD62580	电能图表 光标位置软元件值
	GD62581	电能图表 光标位置软元件值
	GD62582	综合电流图表 日志 ID 软元件
	GD62583	综合电流图表 图表偏置软元件
	GD62584	综合电流图表 上限值软元件
	GD62586	综合电流图表 光标位置时间
	GD62587	综合电流图表 光标位置时间
	GD62588	综合电流图表 光标位置时间
	GD62589	综合电流图表 光标位置时间
	GD62590	综合电流图表 显示开始位置时间
	GD62591	综合电流图表 显示开始位置时间
	GD62592	综合电流图表 显示开始位置时间
	GD62593	综合电流图表 显示开始位置时间
	GD62594	综合电流图表 显示结束位置时间
	GD62595	综合电流图表 显示结束位置时间
	GD62596	综合电流图表 显示结束位置时间
	GD62597	综合电流图表 显示结束位置时间
	GD62598	综合电流图表 光标位置软元件值
	GD62600	日志定时器参数
	GD62601	相线系统 写入工件软元件
	GD62602	传感器类型 写入工件软元件
	GD62603	电流乘比率 写入工件软元件
	GD62604	电压乘比率 写入工件软元件
	GD62605	功率乘比率 写入工件软元件
	GD62606	电能乘比率 写入工件软元件
	GD62607	参数写入处理选择软元件
	GD62608	相线系统 显示软元件
	GD62609	电流乘比率 显示软元件
	GD62610	电压乘比率 显示软元件
	GD62611	功率乘比率 显示软元件
	GD62612	电能乘比率 显示软元件
	GD62613	报警显示(用户) 注释号
	GD62614	报警显示(用户) 发生日期
	GD62616	报警显示(用户) 发生时间
	GD62618	报警显示(用户) 恢复日期
	GD62620	报警显示(用户) 恢复时间
	GD62622	选择站号位指示灯的偏置软元件
	GD62623	站号选择窗口上名称的偏置软元件
	GD62628	一次电压(线电压) 偏置软元件

类型	软元件编号	用途
字	GD62630	一次电压(相电压) 偏置软元件
	GD62632	电能图表 显示位置时间指定软元件(年月)
	GD62633	电能图表 显示位置时间指定软元件(日时)
	GD62634	电能图表 显示位置时间指定软元件(分秒)
	GD62635	综合电流图表 显示位置时间指定软元件(年月)
	GD62636	综合电流图表 显示位置时间指定软元件(日时)
	GD62637	综合电流图表 显示位置时间指定软元件(分秒)
	GD62638	报警显示(用户) 全部站点/监视站显示切换软元件
	GD62640	软元件数据传送 ID: 193 外部控制软元件
	GD62641	软元件数据传送 ID: 193 外部通知软元件
	GD62642	软元件数据传送 ID: 194 外部控制软元件
	GD62643	软元件数据传送 ID: 194 外部通知软元件
	GD62644	软元件数据传送 ID: 195 外部控制软元件
	GD62645	软元件数据传送 ID: 195 外部通知软元件
	GD62646	软元件数据传送 ID: 196 外部控制软元件
	GD62647	软元件数据传送 ID: 196 外部通知软元件
	GD62648	软元件数据传送 ID: 197 外部控制软元件
	GD62649	软元件数据传送 ID: 197 外部通知软元件
	GD62650	软元件数据传送 ID: 198 外部控制软元件
	GD62651	软元件数据传送 ID: 198 外部通知软元件
	GD62652	软元件数据传送 ID: 199 外部控制软元件
	GD62653	软元件数据传送 ID: 199 外部通知软元件
	GD62654	软元件数据传送 ID: 200 外部控制软元件
	GD62655	软元件数据传送 ID: 200 外部通知软元件
	GD62656	软元件数据传送 ID: 201 外部控制软元件
	GD62657	软元件数据传送 ID: 201 外部通知软元件
	GD62658	软元件数据传送 ID: 202 外部控制软元件
	GD62659	软元件数据传送 ID: 202 外部通知软元件
	GD62660	软元件数据传送 ID: 203 外部控制软元件
	GD62661	软元件数据传送 ID: 203 外部通知软元件
	GD62662	软元件数据传送 ID: 204 外部控制软元件
	GD62663	软元件数据传送 ID: 204 外部通知软元件
	GD62664	软元件数据传送 ID: 205 外部控制软元件
	GD62665	软元件数据传送 ID: 205 外部通知软元件
	GD62666	软元件数据传送 ID: 206 外部控制软元件
	GD62667	软元件数据传送 ID: 206 外部通知软元件
	GD62668	软元件数据传送 ID: 207 外部控制软元件
	GD62669	软元件数据传送 ID: 207 外部通知软元件
	GD62670	软元件数据传送 ID: 208 外部控制软元件
	GD62671	软元件数据传送 ID: 208 外部通知软元件
	GD62672	软元件数据传送 ID: 209 外部控制软元件
	GD62673	软元件数据传送 ID: 209 外部通知软元件
	GD62674	软元件数据传送 ID: 210 外部控制软元件
	GD62675	软元件数据传送 ID: 210 外部通知软元件
	GD62676	软元件数据传送 ID: 211 外部控制软元件
	GD62677	软元件数据传送 ID: 211 外部通知软元件
	GD62678	软元件数据传送 ID: 212 外部控制软元件
	GD62679	软元件数据传送 ID: 212 外部通知软元件
	GD62680	软元件数据传送 ID: 213 外部控制软元件
	GD62681	软元件数据传送 ID: 213 外部通知软元件
	GD62682	软元件数据传送 ID: 214 外部控制软元件
	GD62683	软元件数据传送 ID: 214 外部通知软元件
	GD62684	软元件数据传送 ID: 215 外部控制软元件

类型	软元件编号	用途
字	GD62685	软元件数据传送 ID: 215 外部通知软元件
	GD62686	软元件数据传送 ID: 216 外部控制软元件
	GD62687	软元件数据传送 ID: 216 外部通知软元件
	GD62688	软元件数据传送 ID: 217 外部控制软元件
	GD62689	软元件数据传送 ID: 217 外部通知软元件
	GD62690	软元件数据传送 ID: 218 外部控制软元件
	GD62691	软元件数据传送 ID: 218 外部通知软元件
	GD62692	软元件数据传送 ID: 219 外部控制软元件
	GD62693	软元件数据传送 ID: 219 外部通知软元件
	GD62694	软元件数据传送 ID: 220 外部控制软元件
	GD62695	软元件数据传送 ID: 220 外部通知软元件
	GD62696	软元件数据传送 ID: 221 外部控制软元件
	GD62697	软元件数据传送 ID: 221 外部通知软元件
	GD62698	软元件数据传送 ID: 222 外部控制软元件
	GD62699	软元件数据传送 ID: 222 外部通知软元件
	GD62700	软元件数据传送 ID: 223 外部控制软元件
	GD62701	软元件数据传送 ID: 223 外部通知软元件
	GD62702	软元件数据传送 ID: 224 外部控制软元件
	GD62703	软元件数据传送 ID: 224 外部通知软元件
	GD62704	软元件数据传送 ID: 225 外部控制软元件
	GD62705	软元件数据传送 ID: 225 外部通知软元件
	GD62706	软元件数据传送 ID: 226 外部控制软元件
	GD62707	软元件数据传送 ID: 226 外部通知软元件
	GD62708	软元件数据传送 ID: 227 外部控制软元件
	GD62709	软元件数据传送 ID: 227 外部通知软元件
	GD62710	软元件数据传送 ID: 228 外部控制软元件
	GD62711	软元件数据传送 ID: 228 外部通知软元件
	GD62712	软元件数据传送 ID: 229 外部控制软元件
	GD62713	软元件数据传送 ID: 229 外部通知软元件
	GD62714	软元件数据传送 ID: 230 外部控制软元件
	GD62715	软元件数据传送 ID: 230 外部通知软元件
	GD62716	软元件数据传送 ID: 231 外部控制软元件
	GD62717	软元件数据传送 ID: 231 外部通知软元件
	GD62718	软元件数据传送 ID: 232 外部控制软元件
	GD62719	软元件数据传送 ID: 232 外部通知软元件
	GD62720	软元件数据传送 ID: 233 外部控制软元件
	GD62721	软元件数据传送 ID: 233 外部通知软元件
	GD62722	软元件数据传送 ID: 234 外部控制软元件
	GD62723	软元件数据传送 ID: 234 外部通知软元件
	GD62724	软元件数据传送 ID: 235 外部控制软元件
	GD62725	软元件数据传送 ID: 235 外部通知软元件
	GD62726	软元件数据传送 ID: 236 外部控制软元件
	GD62727	软元件数据传送 ID: 236 外部通知软元件
	GD62728	软元件数据传送 ID: 237 外部控制软元件
	GD62729	软元件数据传送 ID: 237 外部通知软元件
	GD62730	软元件数据传送 ID: 238 外部控制软元件
	GD62731	软元件数据传送 ID: 238 外部通知软元件
	GD62732	软元件数据传送 ID: 239 外部控制软元件
	GD62733	软元件数据传送 ID: 239 外部通知软元件
	GD62734	软元件数据传送 ID: 240 外部控制软元件
	GD62735	软元件数据传送 ID: 240 外部通知软元件
	GD62736	软元件数据传送 ID: 241 外部控制软元件
	GD62737	软元件数据传送 ID: 241 外部通知软元件

类型	软元件编号	用途
字	GD62738	软元件数据传送 ID: 242 外部控制软元件
	GD62739	软元件数据传送 ID: 242 外部通知软元件
	GD62740	软元件数据传送 ID: 243 外部控制软元件
	GD62741	软元件数据传送 ID: 243 外部通知软元件
	GD62742	软元件数据传送 ID: 244 外部控制软元件
	GD62743	软元件数据传送 ID: 244 外部通知软元件
	GD62744	软元件数据传送 ID: 245 外部控制软元件
	GD62745	软元件数据传送 ID: 245 外部通知软元件
	GD62746	软元件数据传送 ID: 246 外部控制软元件
	GD62747	软元件数据传送 ID: 246 外部通知软元件
	GD62748	软元件数据传送 ID: 247 外部控制软元件
	GD62749	软元件数据传送 ID: 247 外部通知软元件
	GD62750	软元件数据传送 ID: 248 外部控制软元件
	GD62751	软元件数据传送 ID: 248 外部通知软元件
	GD62752	软元件数据传送 ID: 249 外部控制软元件
	GD62753	软元件数据传送 ID: 249 外部通知软元件
	GD62754	软元件数据传送 ID: 250 外部控制软元件
	GD62755	软元件数据传送 ID: 250 外部通知软元件
	GD62756	软元件数据传送 ID: 251 外部控制软元件
	GD62757	软元件数据传送 ID: 251 外部通知软元件
	GD62758	软元件数据传送 ID: 252 外部控制软元件
	GD62759	软元件数据传送 ID: 252 外部通知软元件
	GD62760	软元件数据传送 ID: 253 外部控制软元件
	GD62761	软元件数据传送 ID: 253 外部通知软元件
	GD62762	软元件数据传送 ID: 254 外部控制软元件
	GD62763	软元件数据传送 ID: 254 外部通知软元件
	GD62765	监视对象站选择开关
	GD62766	现在机种保存软元件(站号切换时时机更改用)
	GD63990	时钟的数字开关
	GD63991	时钟的数字开关
	GD63992	时钟的数字开关
	GD63993	时钟的数字开关
	GD63994	时钟的数字开关
	GD63995	时钟的数字开关
	GS531、GS532	监视站切断(CH1)
	GS573	MODBUS®通信协议功能(公共)
	TMP800～TMP803	工程脚本、画面脚本运算用
	TMP950～TMP996	时钟设置画面 对象脚本运算用

5.5 注释一览表

注释组号	注释号	使用处
254	No. 1～319	B-30011 (用户报警监视 ID 30001)
255	No. 1	B-30001
	No. 3、4	B-30002～30011、B-30020
	No. 7	B-30001～30011、B-30020
	No. 10	B-30020、W-30015
	No. 11～17	B-30020
	No. 18	B-30002、B-30020
	No. 20～37	B-30002
	No. 40	B-30002
	No. 47	B-30001、B-30009
	No. 48	B-30001、B-30010
	No. 49	B-30002、B-30009、B-30010、B-30020
	No. 50	B-30009、B-30010
	No. 57	B-30001
	No. 59	B-30001、B-30011
	No. 61、62	B-30010
	No. 63、64	B-30009、B-30010
	No. 68	B-30009
	No. 70、71	B-30001～30011、B-30020
	No. 73	B-30002、B-30020
	No. 75、76	B-30011
	No. 79	B-30020、W-30016
	No. 80	B-30020、W-30017
	No. 81	B-30020、W-30018
	No. 82	B-30020、W-30019
	No. 83	B-30020、W-30020
	No. 84、85	B-30020
	No. 88～94	W-30003
	No. 95	W-30002
	No. 96	W-30003
	No. 98	W-30001
	No. 99	W-30001、W-30003
	No. 101～110	B-30009、B-30010
	No. 210	W-30030、W-30031
	No. 211	W-30031
	No. 212	W-30030、W-30031
	No. 213	W-30031
	No. 215、216	W-30031、W-30015～30020
	No. 217	B-30020
	No. 220	W-30030
	No. 221	W-30031
	No. 222	W-30030
	No. 223	W-30031
	No. 300、301	W-30031
	No. 303～309	W-30015
	No. 310	B-30020、W-30016
	No. 311	W-30016
	No. 312～313	W-30019
	No. 314	W-30017、W-30019、W-30020
	No. 315～317	W-30017～30020
	No. 318	W-30018～30020

注释组号	注释号	使用处
255	No. 319	W-30020
	No. 401	W-30014
	No. 402、403	W-30014~30020
	No. 500	B-30002、B-30009~30011、B-30020
	No. 501	B-30001、B-30020
	No. 502~504	B-30002、B-30009~30011、B-30020

5.6 配方一览表

5.6.1 公共设置

外部通知信息	
外部控制软元件	GD62554
配方号存储软元件	GD62555
记录号存储软元件	GD62556
外部通知软元件	GD62557
配方号通知软元件	GD62558
记录号通知软元件	GD62559

5.6.2 个别设置

配方 No. 30001 配方 1

项目		设置
配方文件	配方文件	使用配方文件(执行写入・读取)
	文件格式	G2P(二进制)
	驱动器名	A: 标准 SD 卡
	文件夹名	Package1
	文件名	ARP30001.G2P
触发软元件	写入触发软元件 1	GD62560.b0
	读取触发软元件 1	GD62560.b1
	记录号软元件	GD62561
块数		65
记录数		1
块 1	软元件	GS531
	软元件类型	有符号 BIN16
	点数	2
块 2	软元件	GD61000
	软元件类型	字符串
	点数	20

项目		设置
块 3	软元件	GD61020
	软元件类型	有符号 BIN16
	点数	20
块 4	软元件	GD61040
	软元件类型	字符串
	点数	20
块 5	软元件	GD61060
	软元件类型	有符号 BIN16
	点数	20
块 6	软元件	GD61080
	软元件类型	字符串
	点数	20
块 7	软元件	GD61100
	软元件类型	有符号 BIN16
	点数	20
块 8	软元件	GD61120
	软元件类型	字符串
	点数	20
块 9	软元件	GD61140
	软元件类型	有符号 BIN16
	点数	20
块 10	软元件	GD61160
	软元件类型	字符串
	点数	20
块 11	软元件	GD61180
	软元件类型	有符号 BIN16
	点数	20
块 12	软元件	GD61200
	软元件类型	字符串
	点数	20
块 13	软元件	GD61220
	软元件类型	有符号 BIN16
	点数	20

项目		设置
块 14	软元件	GD61240
	软元件类型	字符串
	点数	20
块 15	软元件	GD61260
	软元件类型	有符号 BIN16
	点数	20
块 16	软元件	GD61280
	软元件类型	字符串
	点数	20
块 17	软元件	GD61300
	软元件类型	有符号 BIN16
	点数	20
块 18	软元件	GD61320
	软元件类型	字符串
	点数	20
块 19	软元件	GD61340
	软元件类型	有符号 BIN16
	点数	20
块 20	软元件	GD61360
	软元件类型	字符串
	点数	20
块 21	软元件	GD61380
	软元件类型	有符号 BIN16
	点数	20
块 22	软元件	GD61400
	软元件类型	字符串
	点数	20
块 23	软元件	GD61420
	软元件类型	有符号 BIN16
	点数	20
块 24	软元件	GD61440
	软元件类型	字符串
	点数	20

项目		设置
块 25	软元件	GD61460
	软元件类型	有符号 BIN16
	点数	20
块 26	软元件	GD61480
	软元件类型	字符串
	点数	20
块 27	软元件	GD61500
	软元件类型	有符号 BIN16
	点数	20
块 28	软元件	GD61520
	软元件类型	字符串
	点数	20
块 29	软元件	GD61540
	软元件类型	有符号 BIN16
	点数	20
块 30	软元件	GD61560
	软元件类型	字符串
	点数	20
块 31	软元件	GD61580
	软元件类型	有符号 BIN16
	点数	20
块 32	软元件	GD61600
	软元件类型	字符串
	点数	20
块 33	软元件	GD61620
	软元件类型	有符号 BIN16
	点数	20
块 34	软元件	GD61640
	软元件类型	字符串
	点数	20
块 35	软元件	GD61660
	软元件类型	有符号 BIN16
	点数	20

项目		设置
块 36	软元件	GD61680
	软元件类型	字符串
	点数	20
块 37	软元件	GD61700
	软元件类型	有符号 BIN16
	点数	20
块 38	软元件	GD61720
	软元件类型	字符串
	点数	20
块 39	软元件	GD61740
	软元件类型	有符号 BIN16
	点数	20
块 40	软元件	GD61760
	软元件类型	字符串
	点数	20
块 41	软元件	GD61780
	软元件类型	有符号 BIN16
	点数	20
块 42	软元件	GD61800
	软元件类型	字符串
	点数	20
块 43	软元件	GD61820
	软元件类型	有符号 BIN16
	点数	20
块 44	软元件	GD61840
	软元件类型	字符串
	点数	20
块 45	软元件	GD61860
	软元件类型	有符号 BIN16
	点数	20
块 46	软元件	GD61880
	软元件类型	字符串
	点数	20

项目		设置
块 47	软元件	GD61900
	软元件类型	有符号 BIN16
	点数	20
块 48	软元件	GD61920
	软元件类型	字符串
	点数	20
块 49	软元件	GD61940
	软元件类型	有符号 BIN16
	点数	20
块 50	软元件	GD61960
	软元件类型	字符串
	点数	20
块 51	软元件	GD61980
	软元件类型	有符号 BIN16
	点数	20
块 52	软元件	GD62000
	软元件类型	字符串
	点数	20
块 53	软元件	GD62020
	软元件类型	有符号 BIN16
	点数	20
块 54	软元件	GD62040
	软元件类型	字符串
	点数	20
块 56	软元件	GD62060
	软元件类型	有符号 BIN16
	点数	20
块 57	软元件	GD62080
	软元件类型	字符串
	点数	20
块 58	软元件	GD62100
	软元件类型	有符号 BIN16
	点数	20

项目		设置
块 59	软元件	GD62120
	软元件类型	字符串
	点数	20
块 60	软元件	GD62140
	软元件类型	有符号 BIN16
	点数	20
块 61	软元件	GD62160
	软元件类型	字符串
	点数	20
块 62	软元件	GD62180
	软元件类型	有符号 BIN16
	点数	20
块 63	软元件	GD62200
	软元件类型	字符串
	点数	20
块 64	软元件	GD62220
	软元件类型	有符号 BIN16
	点数	20

5.7 脚本一览表

项目	设置
工程脚本	有
画面脚本	B-30001~B-30020、W-30030~W-30031
对象脚本	B-30002~B-30020、W-30003

5.7.1 工程脚本

脚本号.	30001	脚本名	Script30001
注释	画面启动时的处理		
数据类型	有符号 BIN16	触发类型	上升沿 GB40
<pre>//Process when screen opened. //Write the parameters [w:GD62561] = 1; //Record No. set([b:GD62560.b0]); [w:GS573] = 2; //Specify the number of holding registers read.</pre>			

脚本号.	30002	脚本名	Script30002
注释	配方写入完成		
数据类型	有符号 BIN16	触发类型	上升沿 GD62557.b4
<pre>//Process after advanced recipe write. //Find the lowest valid station No. [u32:TMP0800] = [u32:GS531] >> 1; //Because 0 bit is unused, it is shifted to 1 bit to the right. [w:TMP0802] = 1; while([w:TMP0802] < 32) { if(([u32:TMP0800] & 0x00000001) == 0) { [w:GD10] = [w:TMP0802]; break; } [u32:TMP0800] = [u32:TMP0800] >> 1; [w:TMP0802] = [w:TMP0802] + 1; } //Set 1 in case all stations are invalid. if([w:TMP0802] == 32) { [w:GD10] = 1; } //Open the setting for screen startup. set([b:GB61013]); //Clear the flag rst([b:GD62560.b0]);</pre>			

脚本号.	30024	脚本名	Script30024
注释	日志延迟		
数据类型	有符号 BIN16	触发类型	OFF 中周期/2 秒 GB61011
//After starting up the screen, wait for 2 seconds and start the logging script. set([b:GB61011]);			

脚本号.	30007	脚本名	Script30007
注释	日志触发 OFF (No. 30001)		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61000
//Trigger OFF logging No.30001 (electric energy (1 hour)). rst([b:GB61000]);			

脚本号.	30008	脚本名	Script30008
注释	日志触发 OFF (No. 30002)		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61001
//Trigger OFF logging No. 30002 (total current (1 hour)). rst([b:GB61001]);			

脚本号.	30009	脚本名	Script30009
注释	日志触发 OFF (No. 30003)		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61002
//Trigger OFF logging No.30003 (electric energy (1 month)). rst([b:GB61002]);			

脚本号.	30010	脚本名	Script30010
注释	日志触发 OFF (No. 30004)		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61003
//Trigger OFF logging No. 30004 (total current (1 minute)). rst([b:GB61003]);			

脚本号.	30011	脚本名	Script30011
注释	日志触发 OFF (No. 30005)		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61004
//Trigger OFF logging No. 30005 (total current (30 minutes)). rst([b:GB61004]);			

脚本号.	30012	脚本名	Script30012
注释	日志触发 OFF (No. 30006)		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61005
//Trigger OFF logging No. 30006 (total current (1 hour)). rst([b:GB61005]);			

脚本号.	30013	脚本名	Script30013
注释	日志触发 OFF (No. 30007)		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61006
//Trigger OFF logging No. 30007 (total current (1 day)). rst([b:GB61006]);			

脚本号.	30014	脚本名	Script30014
注释	输入完成后，读出到配方		
数据类型	有符号 BIN16	触发类型	上升沿 GB61009
<pre>//Read advanced recipe for each object after input. set([b:GD62560.b1]);</pre>			

脚本号.	30015	脚本名	Script30015
注释	配方读出完成后处理		
数据类型	有符号 BIN16	触发类型	ON 中 GD62557.b5
<pre>//Trigger OFF after advanced recipe read. //After setting valid/invalid station No., //start the script to clear logging data. if([b:GB61015] == ON) { set([b:GB61014]); rst([b:GB61015]); } rst([b:GD62560.b1]); rst([b:GD62557.b5]);</pre>			

脚本号.	30018	脚本名	Script30018
注释	软元件数据传送 1 触发 OFF		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61018
<pre>//Reset with regularly monitoring the trigger for device data transfer. rst([b:GD62640.b0]); rst([b:GD62644.b0]); rst([b:GD62648.b0]); rst([b:GD62652.b0]); rst([b:GD62656.b0]); rst([b:GD62660.b0]); rst([b:GD62664.b0]); rst([b:GD62668.b0]); rst([b:GD62672.b0]); rst([b:GD62676.b0]); rst([b:GD62680.b0]); rst([b:GD62684.b0]); rst([b:GD62688.b0]); rst([b:GD62692.b0]); rst([b:GD62696.b0]); rst([b:GD62700.b0]); rst([b:GD62704.b0]); rst([b:GD62708.b0]); rst([b:GD62712.b0]); rst([b:GD62716.b0]); rst([b:GD62720.b0]); rst([b:GD62724.b0]); rst([b:GD62728.b0]); rst([b:GD62732.b0]); rst([b:GD62736.b0]); rst([b:GD62740.b0]); rst([b:GD62744.b0]); rst([b:GD62748.b0]); rst([b:GD62752.b0]);</pre>			

```

rst([b:GD62756.b0]);
rst([b:GD62760.b0]);

rst([b:GD62642.b0]);
rst([b:GD62646.b0]);
rst([b:GD62650.b0]);
rst([b:GD62654.b0]);
rst([b:GD62658.b0]);
rst([b:GD62662.b0]);
rst([b:GD62666.b0]);
rst([b:GD62670.b0]);
rst([b:GD62674.b0]);
rst([b:GD62678.b0]);
rst([b:GD62682.b0]);
rst([b:GD62686.b0]);
rst([b:GD62690.b0]);
rst([b:GD62694.b0]);
rst([b:GD62698.b0]);
rst([b:GD62702.b0]);
rst([b:GD62706.b0]);
rst([b:GD62710.b0]);
rst([b:GD62714.b0]);
rst([b:GD62718.b0]);
rst([b:GD62722.b0]);
rst([b:GD62726.b0]);
rst([b:GD62730.b0]);
rst([b:GD62734.b0]);
rst([b:GD62738.b0]);
rst([b:GD62742.b0]);
rst([b:GD62746.b0]);
rst([b:GD62750.b0]);
rst([b:GD62754.b0]);
rst([b:GD62758.b0]);
rst([b:GD62762.b0]);

rst([b:GB61018]);

```

脚本号.	30026	脚本名	Script30026
注释	软元件数据传送 2 触发 OFF		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GD62642. b0
//Reset with regularly monitoring the trigger for device data transfer.			
<pre> rst([b:GD62642.b0]); rst([b:GD62646.b0]); rst([b:GD62650.b0]); rst([b:GD62654.b0]); rst([b:GD62658.b0]); rst([b:GD62662.b0]); rst([b:GD62666.b0]); rst([b:GD62670.b0]); rst([b:GD62674.b0]); rst([b:GD62678.b0]); rst([b:GD62682.b0]); rst([b:GD62686.b0]); rst([b:GD62690.b0]); rst([b:GD62694.b0]); </pre>			


```

rst([b:GD62698.b0]);
rst([b:GD62702.b0]);
rst([b:GD62706.b0]);
rst([b:GD62710.b0]);
rst([b:GD62714.b0]);
rst([b:GD62718.b0]);
rst([b:GD62722.b0]);
rst([b:GD62726.b0]);
rst([b:GD62730.b0]);
rst([b:GD62734.b0]);
rst([b:GD62738.b0]);
rst([b:GD62742.b0]);
rst([b:GD62746.b0]);
rst([b:GD62750.b0]);
rst([b:GD62754.b0]);
rst([b:GD62758.b0]);
rst([b:GD62762.b0]);

```

5.7.2 画面脚本

基本画面公共

脚本号.	30004	脚本名	Script30004
注释	日志相关 1: 软件件数据传送开始		
数据类型	有符号 BIN16	触发类型	ON 中 GB61011
<pre> //Obtain data for logging by device data transfer. //1 分每 if(((([w:GS652] & 0x00FF) == 0) && ([b:GB61017] == OFF)) { set([b:GB61017]); //Flag not to execute several times in one second. //Every 1 hours if(((([w:GS652] & 0xFF00) >> 8) == 0) { //Every 1 days if(([w:GS651] & 0x00FF) == 0) { //Every 1 months if(((([w:GS651] & 0xFF00) >> 8) == 0x0001) { //Start device data transfer [w:GD62600] = 5; //Execute device data transfer for only validate stations. [u32:TMP0800] = [u32:GS531] >> 1; //Because 0 bit is unused, it is shifted to 1 bit to the right. [w:TMP0802] = 1; //The number of loop (also as station No.)) [w:TMP0803] = 0; //Offset to check model. [w:TMP0804] = 0; //Device data transfer start trigger offset. while([w:TMP0802] < 32) { if(([u32:TMP0800] & 0x00000001) == 0) { [w:TMP0804] = ([w:TMP0802] - 1) * 4; //Model check if([w:GD61020[w:TMP0803]] < 1) </pre>			

```

        {
            set([b:GD62640. b0[w:TMP0804]]);
        }else{
            set([b:GD62642. b0[w:TMP0804]]);
        }
    }

    [u32:TMP0800] = [u32:TMP0800] >> 1;
    [w:TMP0802] = [w:TMP0802] + 1;
    [w:TMP0803] = [w:TMP0803] + 40;
}
set([b:GB61007]);
return;
}

//Start device data transfer
[w:GD62600] = 4;

//Execute device data transfer for only validate stations.
[u32:TMP0800] = [u32:GS531] >> 1;    //Because 0 bit is unused, it is shifted to 1 bit to the
right.

[w:TMP0802] = 1; //The number of loop (also as station No.)
[w:TMP0803] = 0; //Offset to check model.
[w:TMP0804] = 0; //Device data transfer start trigger offset.

while([w:TMP0802] < 32)
{
    if(([u32:TMP0800] & 0x00000001) == 0)
    {
        [w:TMP0804] = ([w:TMP0802] - 1) * 4;

        //Model check
        if([w:GD61020[w:TMP0803]] < 1)
        {
            set([b:GD62640. b0[w:TMP0804]]);
        }else{
            set([b:GD62642. b0[w:TMP0804]]);
        }
    }
}

    [u32:TMP0800] = [u32:TMP0800] >> 1;
    [w:TMP0802] = [w:TMP0802] + 1;
    [w:TMP0803] = [w:TMP0803] + 40;
}
set([b:GB61007]);
return;
}

//Start device data transfer
[w:GD62600] = 3;

//Execute device data transfer for only validate stations.
[u32:TMP0800] = [u32:GS531] >> 1;    //Because 0 bit is unused, it is shifted to 1 bit to the
right.

```

```

[w:TMP0802] = 1; //The number of loop (also as station No.)
[w:TMP0803] = 0; //Offset to check model.
[w:TMP0804] = 0; //Device data transfer start trigger offset.

while([w:TMP0802] < 32)
{
    if(([u32:TMP0800] & 0x00000001) == 0)
    {
        [w:TMP0804] = ([w:TMP0802] - 1) * 4;

        //Model check
        if([w:GD61020[w:TMP0803]] < 1)
        {
            set([b:GD62640.b0[w:TMP0804]]);
        }else{
            set([b:GD62642.b0[w:TMP0804]]);
        }
    }

    [u32:TMP0800] = [u32:TMP0800] >> 1;
    [w:TMP0802] = [w:TMP0802] + 1;
    [w:TMP0803] = [w:TMP0803] + 40;
}
set([b:GB61007]);
return;

}else{
    //Every 30 minutes
    if((([w:GS652] & 0xFF00) >> 8) == 0x0030)
    {
        //Start device data transfer
        [w:GD62600] = 2;

        //Execute device data transfer for only validate stations.
        [u32:TMP0800] = [u32:GS531] >> 1;    //Because 0 bit is unused, it is shifted to 1 bit to the
right.

        [w:TMP0802] = 1; //The number of loop (also as station No.)
        [w:TMP0803] = 0; //Offset to check model.
        [w:TMP0804] = 0; //Device data transfer start trigger offset.

        while([w:TMP0802] < 32)
        {
            if(([u32:TMP0800] & 0x00000001) == 0)
            {
                [w:TMP0804] = ([w:TMP0802] - 1) * 4;

                //Model check
                if([w:GD61020[w:TMP0803]] < 1)
                {
                    set([b:GD62640.b0[w:TMP0804]]);
                }else{
                    set([b:GD62642.b0[w:TMP0804]]);
                }
            }
        }
    }
}

```

```

        [u32:TMP0800] = [u32:TMP0800] >> 1;
        [w:TMP0802] = [w:TMP0802] + 1;
        [w:TMP0803] = [w:TMP0803] + 40;
    }
    set([b:GB61007]);
    return;
}

//Start device data transfer
[w:GD62600] = 1;

//Execute device data transfer for only validate stations.
[u32:TMP0800] = [u32:GS531] >> 1;    //Because 0 bit is unused, it is shifted to 1 bit to the right.

[w:TMP0802] = 1; //The number of loop (also as station No.)
[w:TMP0803] = 0; //Offset to check model.
[w:TMP0804] = 0; //Device data transfer start trigger offset.

while([w:TMP0802] < 32)
{
    if(([u32:TMP0800] & 0x00000001) == 0)
    {
        [w:TMP0804] = ([w:TMP0802] - 1) * 4;

        //Model check
        if([w:GD61020[w:TMP0803]] < 1)
        {
            set([b:GD62640.b0[w:TMP0804]]);
        }else{
            set([b:GD62642.b0[w:TMP0804]]);
        }
    }

    [u32:TMP0800] = [u32:TMP0800] >> 1;
    [w:TMP0802] = [w:TMP0802] + 1;
    [w:TMP0803] = [w:TMP0803] + 40;
}
set([b:GB61007]);
return;
}else{
    if(((([w:GS652] & 0x00FF) != 0) && ([b:GB61017] == ON))
    {
        rst([b:GB61017]); //Reset the flag after 00 sec.
    }
}
}

```

脚本号.	30005	脚本名	Script30005
注释	日志相关 2: 综合电流的运算		
数据类型	有符号 BIN16	触发类型	ON 中 GB61007

```

//Calculate total current data device-data-transferred.

```

```

//Start after completion of all device data transfer.

```

```

if([u32:GS531] == (~[u32:GD62562] - 1))
{

```

```

    [w:TMP0800] = 0;

```

```

[w:TMP0801] = 0;
[w:TMP0802] = 0;

while([w:TMP0800] < 31)
{
    if([w:GD61020[w:TMP0801]] == 0)
    {
        //EMU4
        switch([s16:GD62457[w:TMP0800]])
        {
            case -3: [flt:GD62302[w:TMP0802]] = [s16:GD62426[w:TMP0800]] * 0.001;
                    break;

            case -2: [flt:GD62302[w:TMP0802]] = [s16:GD62426[w:TMP0800]] * 0.01;
                    break;

            case -1: [flt:GD62302[w:TMP0802]] = [s16:GD62426[w:TMP0800]] * 0.1;
                    break;

            case 0: [flt:GD62302[w:TMP0802]] = [s16:GD62426[w:TMP0800]];
                   break;
        }
    }
    else{
        //ME110
        switch([w:GD61025[w:TMP0801]])
        {
            case 1: [flt:GD62302[w:TMP0802]] = [s16:GD62426[w:TMP0800]] * 0.01 ;
                   break;

            case 2: [flt:GD62302[w:TMP0802]] = [s16:GD62426[w:TMP0800]] * 0.1;
                   break;

            case 3: [flt:GD62302[w:TMP0802]] = [s16:GD62426[w:TMP0800]];
                   break;

            case 4: [flt:GD62302[w:TMP0802]] = [s16:GD62426[w:TMP0800]] * 10;
                   break;
        }
    }

    [w:TMP0800] = [w:TMP0800] + 1;
    [w:TMP0801] = [w:TMP0801] + 25;
    [w:TMP0802] = [w:TMP0802] + 2;
}

//Select the logging type.
switch([w:GD62600])
{
    case 1: set([b:GB61003]); //Every 1 minutes
           break;

    case 2: set([b:GB61003]); //Every 1 minutes
           set([b:GB61004]); //Every 30 minutes
}

```

```

        break;

    case 3:    set([b:GB61003]); //Every 1 minutes
               set([b:GB61004]); //Every 30 minutes
               set([b:GB61005]); //Every 1 hours
               break;

    case 4:    set([b:GB61003]); //Every 1 minutes
               set([b:GB61004]); //Every 30 minutes
               set([b:GB61005]); //Every 1 hours
               set([b:GB61006]); //Every 1 days
               break;

    case 5:    set([b:GB61003]); //Every 1 minutes
               set([b:GB61004]); //Every 30 minutes
               set([b:GB61005]); //Every 1 hours
               set([b:GB61006]); //Every 1 days
               break;

    default:   break;
}

[u32:GD62562] = 0;

set([b:GB61008]); //Operation script for electric energy.

rst([b:GB61007]);
}

```

脚本号.	30006	脚本名	Script30006
注释	日志相关 2: 电能的运算		
数据类型	有符号 BIN16	触发类型	ON 中 GB61008

```

//Calculate electric energy data device-data-transferred.

[w:TMP0800] = 0;
[w:TMP0801] = 0;
[w:TMP0802] = 0;

//Every 1 hours/days/months logging for electric energy.
if([w:GD62600] >= 3)
{
    while([w:TMP0800] <= 31)
    {
        if([w:GD61020[w:TMP0801]] == 0)
        {
            //EMU4
            switch([w:GD62488[w:TMP0800]])
            {
                case -2:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 0.01;
                           break;

                case -1:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 0.1;
                           break;

                case 0:   [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]];
                           break;
            }
        }
    }
}

```

```

        case 1:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 10;
                break;

        case 2:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 100;
                break;
    }
} else {
    //ME110
    switch([w:GD61028[w:TMP0801]])
    {
        case 1:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 0.01 ;
                break;

        case 2:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 0.1;
                break;

        case 3:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]];
                break;

        case 4:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 10;
                break;

        case 5:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 100;
                break;

        case 6:  [flt:GD62240[w:TMP0802]] = [s32:GD62364[w:TMP0800]] * 1000;
                break;
    }
}

[w:TMP0800] = [w:TMP0800] + 1;
[w:TMP0801] = [w:TMP0801] + 25;
[w:TMP0802] = [w:TMP0802] + 2;
}

//Select the logging type.
switch([w:GD62600])
{
    case 3:  set([b:GB61000]); //Every 1 hours
            break;

    case 4:  set([b:GB61000]); //Every 1 hours
            set([b:GB61001]); //Every 1 days
            break;

    case 5:  set([b:GB61000]); //Every 1 hours
            set([b:GB61001]); //Every 1 days
            set([b:GB61002]); //Every 1 months
            break;

    default:  break;
}
}

```

set([b:GB61018]);			
rst([b:GB61008]);			
脚本号.	30021	脚本名	Script30021
注释	将各种参数传送到工作区		
数据类型	有符号 BIN16	触发类型	ON 中 GB61013
<pre>//Transfer the value from recipe to work area of each parameter. ////Current multiplying factor [w:GD62609] = [w:GD61025[w:GD62552]]; ////Voltage multiplying factor [w:GD62610] = [w:GD61026[w:GD62552]]; ////Electric power multiplying factor. [w:GD62611] = [w:GD61027[w:GD62552]]; ////Electric energy multiplying factor. [w:GD62612] = [w:GD61028[w:GD62552]]; ////Graph maximum value (electric energy). [f1t:GD62566] = [f1t:GD61031[w:GD62552]]; ////Maximum graph value (total current) [f1t:GD62584] = ([1-248:s32:400520] * 0.1) * 1.1; ////Turn ON the internal devices (GB61021 or higher) according to the validated station No. [u32:TMP0800] = [u32:GS531] >> 1; //Because 0 bit is unused, it is shifted to 1 bit to the right. [w:TMP0802] = 0; while([w:TMP0802] < 31) { if(([u32:TMP0800] & 0x00000001) == 1) { set([b:GB61021[w:TMP0802]]); } [u32:TMP0800] = [u32:TMP0800] >> 1; [w:TMP0802] = [w:TMP0802] + 1; } //Turn ON the bit lamp to display in Station No. selection window. [w:TMP0803] = [w:GD10] - 1; set([b:GB61060[w:TMP0803]]); rst([b:GB61013]);</pre>			
脚本号.	30027	脚本名	Script30027
注释	一次电流(线电压、相电压)显示/隐藏设置		
数据类型	有符号 BIN16	触发类型	通常
<pre>//Change the display state of the parameters of primary current (line voltage, phase voltage) according to the conditions. if([1-248:w:400513] == 4) { set([b:GB61016]); [w:GD62628] = 2; //Line voltage offset</pre>			

<pre> [w:GD62630] = 0; //Phase voltage offset }else{ rst([b:GB61016]); [w:GD62628] = 0; //Line voltage offset [w:GD62630] = 2; //Phase voltage offset } </pre>			
脚本号.	30039	脚本名	Script30039
注释	站号切换时机更改		
数据类型	有符号 BIN16	触发类型	ON 中周期/1 秒 GB61093
<pre> [b:GB61093]=OFF; [s16:GD63996]=[s16:GD61020[s16:GD62552]]; //Change the timing for station No. switching for actual access according to the model. //The following processes are designed for MEU4. if([s16:GD61020[s16:GD62552]]==1){ switch([w:TMP0800]) { case 1: [w:GD10] = 1; //Station No.1 break; case 2: [w:GD10] = 2; //Station No.2 break; case 3: [w:GD10] = 3; //Station No.3 break; case 4: [w:GD10] = 4; //Station No.4 break; case 5: [w:GD10] = 5; //Station No.5 break; case 6: [w:GD10] = 6; //Station No.6 break; case 7: [w:GD10] = 7; //Station No.7 break; case 8: [w:GD10] = 8; //Station No.8 break; case 9: [w:GD10] = 9; //Station No.9 break; case 10: [w:GD10] = 10; //Station No.10 break; case 11: [w:GD10] = 11; //Station No.11 break; case 12: [w:GD10] = 12; //Station No.12 break; case 13: [w:GD10] = 13; //Station No.13 </pre>			

```

        break;

    case 14:  [w:GD10] = 14;      //Station No.14
        break;

    case 15:  [w:GD10] = 15;      //Station No.15
        break;

    case 16:  [w:GD10] = 16;      //Station No.16
        break;

    case 17:  [w:GD10] = 17;      //Station No.17
        break;

    case 18:  [w:GD10] = 18;      //Station No.18
        break;

    case 19:  [w:GD10] = 19;      //Station No.19
        break;

    case 20:  [w:GD10] = 20;      //Station No.20
        break;

    case 21:  [w:GD10] = 21;      //Station No.21
        break;

    case 22:  [w:GD10] = 22;      //Station No.22
        break;

    case 23:  [w:GD10] = 23;      //Station No.23
        break;

    case 24:  [w:GD10] = 24;      //Station No.24
        break;

    case 25:  [w:GD10] = 25;      //Station No.25
        break;

    case 26:  [w:GD10] = 26;      //Station No.26
        break;

    case 27:  [w:GD10] = 27;      //Station No.27
        break;

    case 28:  [w:GD10] = 28;      //Station No.28
        break;

    case 29:  [w:GD10] = 29;      //Station No.29
        break;

    case 30:  [w:GD10] = 30;      //Station No.30
        break;

    case 31:  [w:GD10] = 31;      //Station No.31
        break;

```

```

    default:    break;
}
}

```

基本画面 30001

脚本号.	30025	脚本名	Script30025
注释	无效站日志数据清除		
数据类型	有符号 BIN16	触发类型	ON 中 GB61014
<pre> //Clear logging data of invalid station No. [u32:TMP0800] = [u32:GS531] >> 1; //Because 0 bit is unused, it is shifted to 1 bit to the right. [w:TMP0802] = 1; [w:TMP0803] = 0; while([w:TMP0802] < 32) { if(([u32:TMP0800] & 0x00000001) == 1) { [flt:GD62240[w:TMP0803]] = 0; //Electric energy [flt:GD62302[w:TMP0803]] = 0; //Total current } [u32:TMP0800] = [u32:TMP0800] >> 1; [w:TMP0802] = [w:TMP0802] + 1; [w:TMP0803] = [w:TMP0803] + 2; } rst([b:GB61014]); </pre>			

基本画面 30009

脚本号.	30017	脚本名	Script30017
注释	功率图表 上限值保存		
数据类型	有符号 BIN16	触发类型	上升沿 GB61010
<pre> //Save the upper limit of electric energy graph [flt:GD61031[w:GD62552]] = [flt:GD62566]; set([b:GD62560.b1]); </pre>			
脚本号.	30029	脚本名	Script30029
注释	当前时间取得		
数据类型	有符号 BIN16	触发类型	上升沿 GB40
<pre> //Store year, month, day, hour, minute, sec when screen is displayed. [w:GD62632]=[w:GS650]; [w:GD62633]=[w:GS651]; [w:GD62634]=[w:GS652]; </pre>			

基本画面 30010

脚本号.	30028	脚本名	Script30028
注释	当前时间取得		
数据类型	有符号 BIN16	触发类型	上升沿 GB40
<pre> //Store year, month, day, hour, minute, sec when screen is displayed. [w:GD62635]=[w:GS650]; [w:GD62636]=[w:GS651]; [w:GD62637]=[w:GS652]; </pre>			

Script No.	30016	脚本名	Script30016
注释	综合电流图表 上限值保存		
数据类型	有符号 BIN16	触发类型	上升沿 GB61010
<pre>//Save the upper limit of average current graph [flt:GD61033[w:GD62552]] = [flt:GD62584]; set([b:GD62560.b1]);</pre>			

基本画面 30020

脚本号.	30020	脚本名	Script30020
注释	参数写入		
数据类型	有符号 BIN16	触发类型	ON 中 GB61012
<pre>//Write parameter in EcoMonitor according to the flag. switch([w:GD62607]) { case 1://Write the value in phase wire device (400513). [1-248:w:400513] = [w:GD62601]; break; case 2://Write the value in sensor type device (400513). [1-248:w:400532] = [w:GD62602]; break; case 3://Write the value in current multiplying factor (GD61011). [w:GD61025[w:GD62552]] = [w:GD62603]; [w:GD62609] = [w:GD62603]; break; case 4://Write the value in voltage multiplying factor (GD61012). [w:GD61026[w:GD62552]] = [w:GD62604]; [w:GD62610] = [w:GD62604]; break; case 5://Set electric power multiplying factor device (GD61013) to value. [w:GD61027[w:GD62552]] = [w:GD62605]; [w:GD62611] = [w:GD62605]; break; case 6://Set electric energy multiplying factor device (GD61014) to value. [w:GD61028[w:GD62552]] = [w:GD62606]; [w:GD62612] = [w:GD62606]; break; default: break; } set([b:GD62560.b1]); rst([b:GB61012]);</pre>			

窗口画面 30030

脚本号.	30037	脚本名	Script30037
注释	W-30008 画面启动时		
数据类型	有符号 BIN16	触发类型	上升沿 GB40
<p>Reflect each setting when window opened.</p> <pre> [w:GD62622] = 0; //Bit lamp offset [w:GD62623] = 0; //Name offset rst([b:GB61092]); //Flag not to display the station No.32. //Trigger OFF only displayed and validated 1 to 8 stations. //OFF=available ON=not available [w:TMP0800] = 0; while([w:TMP0800] < 8) { if([b:GB61021[w:TMP800]] == OFF) { rst([b:GB61052[w:TMP800]]); //Trigger OFF }else{ set([b:GB61052[w:TMP800]]); //Trigger ON } [w:TMP0800] = [w:TMP0800] + 1; } </pre>			
脚本号.	30034	脚本名	Script30034
注释	滚动		
数据类型	有符号 BIN16	触发类型	上升沿 GB61019
<p>//Scroll the screen.</p> <pre> //Set bit lamp offset //Make scrolling loop if([w:GD62622] >= 8) { [w:GD62622] = [w:GD62622] - 8; }else{ [w:GD62622] = 24; } //Set name offset [w:GD62623] = [w:GD62622] * 40; //Trigger OFF only displayed and validated stations. //OFF=available ON=not available [w:TMP0800] = [w:GD62622]; [w:TMP0801] = 0; while([w:TMP0801] < 8) { if([w:TMP0800] != 31) { rst([b:GB61092]); if([b:GB61021[w:TMP800]] == OFF) { rst([b:GB61052[w:TMP0801]]); //Trigger OFF }else{ set([b:GB61052[w:TMP0801]]); //Trigger ON } } } </pre>			

```

    }
  }else{
    set([b:GB61092]);
  }

  [w:TMP0800] = [w:TMP0800] + 1;
  [w:TMP0801] = [w:TMP0801] + 1;
}

```

脚本号.	30035	脚本名	Script30035
注释	滚动		
数据类型	有符号 BIN16	触发类型	上升沿 GB61020

```

//Scroll the screen.

//Set bit lamp offset
//Make scrolling loop
if([w:GD62622] < 24)
{
  [w:GD62622] = [w:GD62622] + 8;
}else{
  [w:GD62622] = 0;
}

//Set name offset
[w:GD62623] = [w:GD62622] * 40;

//Trigger OFF only displayed and validated stations.
//OFF=available ON=not available
[w:TMP0800] = [w:GD62622];
[w:TMP0801] = 0;
while([w:TMP0801] < 8)
{
  if([w:TMP0800] != 31)
  {
    rst([b:GB61092]);

    if([b:GB61021[w:TMP800]] == OFF)
    {
      rst([b:GB61052[w:TMP0801]]); //Trigger OFF
    }else{
      set([b:GB61052[w:TMP0801]]); //Trigger ON
    }
  }else{
    set([b:GB61092]);
  }

  [w:TMP0800] = [w:TMP0800] + 1;
  [w:TMP0801] = [w:TMP0801] + 1;
}

```

脚本号.	30036	脚本名	Script30036
注释	站号切换处理		
数据类型	有符号 BIN16	触发类型	ON 中 GB61091
<pre>//Switch station No. //Also, do the necessary settings to switch the station No. [w:TMP0800] = [w:GD62765] + [w:GD62622]; //Calculate selected station No. [w:TMP0801] = 0; //Turn ON the bit according to the selected station No. while([w:TMP0801] < 31) { if(([w:TMP0801] + 1) == [w:TMP0800]) { set([b:GB61060[w:TMP0801]]); }else{ rst([b:GB61060[w:TMP0801]]); } [w:TMP0801] = [w:TMP0801] + 1; } rst([b:GB61091]);</pre>			
脚本号.	30038	脚本名	Script30038
注释	站号切换		
数据类型	有符号 BIN16	触发类型	关闭画面时
<pre>//Process according to the selected station No. [w:TMP0801] = 0; while([w:TMP0801] < 31) { if([b:GB61060[w:TMP0801]] == ON) { [w:TMP0800] = [w:TMP0801] + 1; break; } [w:TMP0801] = [w:TMP0801] + 1; } //Set the offset value according to station No. switch([w:TMP0800]) { case 1: [w:GD62552] = 0; [w:GD62565] = 0; break; case 2: [w:GD62552] = 40; [w:GD62565] = 2; break; case 3: [w:GD62552] = 80; [w:GD62565] = 4; break; case 4: [w:GD62552] = 120; [w:GD62565] = 6; break;</pre>			

```
case 5:  [w:GD62552] = 160;
        [w:GD62565] = 8;
        break;

case 6:  [w:GD62552] = 200;
        [w:GD62565] = 10;
        break;

case 7:  [w:GD62552] = 240;
        [w:GD62565] = 12;
        break;

case 8:  [w:GD62552] = 280;
        [w:GD62565] = 14;
        break;

case 9:  [w:GD62552] = 320;
        [w:GD62565] = 16;
        break;

case 10: [w:GD62552] = 360;
        [w:GD62565] = 18;
        break;

case 11: [w:GD62552] = 400;
        [w:GD62565] = 20;
        break;

case 12: [w:GD62552] = 440;
        [w:GD62565] = 22;
        break;

case 13: [w:GD62552] = 480;
        [w:GD62565] = 24;
        break;

case 14: [w:GD62552] = 520;
        [w:GD62565] = 26;
        break;

case 15: [w:GD62552] = 560;
        [w:GD62565] = 28;
        break;

case 16: [w:GD62552] = 600;
        [w:GD62565] = 30;
        break;

case 17: [w:GD62552] = 640;
        [w:GD62565] = 32;
        break;

case 18: [w:GD62552] = 680;
        [w:GD62565] = 34;
        break;
```



```

case 19:  [w:GD62552] = 720;
          [w:GD62565] = 36;
          break;

case 20:  [w:GD62552] = 760;
          [w:GD62565] = 38;
          break;

case 21:  [w:GD62552] = 800;
          [w:GD62565] = 40;
          break;

case 22:  [w:GD62552] = 840;
          [w:GD62565] = 42;
          break;

case 23:  [w:GD62552] = 880;
          [w:GD62565] = 44;
          break;

case 24:  [w:GD62552] = 920;
          [w:GD62565] = 46;
          break;

case 25:  [w:GD62552] = 960;
          [w:GD62565] = 48;
          break;

case 26:  [w:GD62552] = 1000;
          [w:GD62565] = 50;
          break;

case 27:  [w:GD62552] = 1040;
          [w:GD62565] = 52;
          break;

case 28:  [w:GD62552] = 1080;
          [w:GD62565] = 54;
          break;

case 29:  [w:GD62552] = 1120;
          [w:GD62565] = 56;
          break;

case 30:  [w:GD62552] = 1160;
          [w:GD62565] = 58;
          break;

case 31:  [w:GD62552] = 1200;
          [w:GD62565] = 60;
          break;

default:  break;
}

```

```

//Change the timing for station No. switching for actual access according to the model.
//The following processes are designed for ME110.
if([s16:GD61020[s16:GD62552]]==0) {
  switch([w:TMP0800])
  {
    case 1:  [w:GD10] = 1;      //Station No.1
            break;

    case 2:  [w:GD10] = 2;      //Station No.2
            break;

    case 3:  [w:GD10] = 3;      //Station No.3
            break;

    case 4:  [w:GD10] = 4;      //Station No.4
            break;

    case 5:  [w:GD10] = 5;      //Station No.5
            break;

    case 6:  [w:GD10] = 6;      //Station No.6
            break;

    case 7:  [w:GD10] = 7;      //Station No.7
            break;

    case 8:  [w:GD10] = 8;      //Station No.8
            break;

    case 9:  [w:GD10] = 9;      //Station No.9
            break;

    case 10: [w:GD10] = 10;     ////Station No.10
            break;

    case 11: [w:GD10] = 11;     ////Station No.11
            break;

    case 12: [w:GD10] = 12;     ////Station No.12
            break;

    case 13: [w:GD10] = 13;     ////Station No.13
            break;

    case 14: [w:GD10] = 14;      //Station No.14
            break;

    case 15: [w:GD10] = 15;      //Station No.15
            break;

    case 16: [w:GD10] = 16;      //Station No.16
            break;

    case 17: [w:GD10] = 17;      //Station No.17
            break;
  }
}

```

```

case 18: [w:GD10] = 18;    //Station No.18
        break;

case 19: [w:GD10] = 19;    //Station No.19
        break;

case 20: [w:GD10] = 20;    //Station No.20
        break;

case 21: [w:GD10] = 21;    //Station No.1
        break;

case 22: [w:GD10] = 22;    //Station No.22
        break;

case 23: [w:GD10] = 23;    ///tation No.23
        break;

case 24: [w:GD10] = 24;    //Station No.24
        break;

case 25: [w:GD10] = 25;    //Station No.25
        break;

case 26: [w:GD10] = 26;    //Station No.26
        break;

case 27: [w:GD10] = 27;    //Station No.27
        break;
case 28: [w:GD10] = 28;    //Station No.28
        break;

case 29: [w:GD10] = 29;    //Station No.29
        break;

case 30: [w:GD10] = 30;    //Station No.30
        break; //Station No. //Station No.

case 31: [w:GD10] = 31;    //Station No.31
        break;

default: break;
}
}

//Station No. [w:GD62553] = [w:GD61020[w:GD62552]];

//Current multiplying factor [w:GD62609] = [w:GD61025[w:GD62552]];

//Voltage multiplying factor
[w:GD62610] = [w:GD61026[w:GD62552]];

//Electric power multiplying factor.
[w:GD62611] = [w:GD61027[w:GD62552]];

```

```
//Electric energy multiplying factor.  
[w:GD62612] = [w:GD61028[w:GD62552]];  
  
//Electric energy Graph upper limit  
[flt:GD62566] = [flt:GD61031[w:GD62552]];  
  
//Average current Graph upper limit  
[flt:GD62584] = [flt:GD61033[w:GD62552]];  
  
//Added to adjust the timing of trigger for the object script.  
set([b:GB61093]);
```

窗口画面 30031

脚本号.	30033	脚本名	Script30033
注释	W-30004 画面启动时		
数据类型	有符号 BIN16	触发类型	上升沿 GB40
<p>Reflect each setting when window opened.</p> <pre> [w:GD62622] = 0; //Bit lamp offset [w:GD62623] = 0; //Name and model offset rst([b:GB61092]); //Flag not to display the station No.32. //Trigger OFF only displayed and validated 1 to 8 stations. //OFF=available ON=not available [w:TMP0800] = 0; while([w:TMP0800] < 8) { if([b:GB61021[w:TMP800]] == OFF) { rst([b:GB61052[w:TMP0800]]); //Trigger OFF }else{ set([b:GB61052[w:TMP0800]]); //Trigger ON } [w:TMP0800] = [w:TMP0800] + 1; } </pre>			
脚本号.	30031	脚本名	Script30031
注释	滚动		
数据类型	有符号 BIN16	触发类型	上升沿 GB61019
<pre> //Scroll the screen. //Set bit lamp offset //Make scrolling loop if([w:GD62622] >= 8) { [w:GD62622] = [w:GD62622] - 8; }else{ [w:GD62622] = 24; } //Set name offset [w:GD62623] = [w:GD62622] * 40; //Trigger OFF only displayed and validated stations. //OFF=available ON=not available [w:TMP0800] = [w:GD62622]; [w:TMP0801] = 0; while([w:TMP0801] < 8) { if([w:TMP0800] != 31) { rst([b:GB61092]); if([b:GB61021[w:TMP800]] == OFF) { rst([b:GB61052[w:TMP0801]]); //Trigger OFF </pre>			

<pre> }else{ set([b:GB61052[w:TMP0801]]); //Trigger ON } }else{ set([b:GB61092]); } [w:TMP0800] = [w:TMP0800] + 1; [w:TMP0801] = [w:TMP0801] + 1; } </pre>			
脚本号.	30032	脚本名	Script30032
注释	滚动		
数据类型	有符号 BIN16	触发类型	上升沿 GB61020
<pre> //Scroll the screen. //Set bit lamp offset //Make scrolling loop if([w:GD62622] < 24) { [w:GD62622] = [w:GD62622] + 8; }else{ [w:GD62622] = 0; } //Make scrolling loop [w:GD62623] = [w:GD62622] * 40; //Trigger OFF only displayed and validated stations. //OFF=available ON=not available [w:TMP0800] = [w:GD62622]; [w:TMP0801] = 0; while([w:TMP0801] < 8) { if([w:TMP0800] != 31) { rst([b:GB61092]); if([b:GB61021[w:TMP800]] == OFF) { rst([b:GB61052[w:TMP0801]]); //Trigger OFF }else{ set([b:GB61052[w:TMP0801]]); //Trigger ON } }else{ set([b:GB61092]); } [w:TMP0800] = [w:TMP0800] + 1; [w:TMP0801] = [w:TMP0801] + 1; } </pre>			
脚本号.	30030	脚本名	Script30030
注释	W-30004 配方保存		
数据类型	有符号 BIN16	触发类型	关闭画面时
<pre> //Save setting results which were set in W-30004 in recipe. [w:TMP0800] = 0; [u32:TMP0801] = [u32:GS531]; //Store the latest current station block. </pre>			

```
//Reflect ON/OFF status which were set in W-33004 to the word device.
while([w:TMP0800] < 32)
{
  if([b:GB61021[w:TMP0800]] == ON)
  {
    [u32:TMP0803] = 0x00000002 << [w:TMP0800] ;
    [u32:TMP0801] = [u32:TMP0801] | [u32:TMP0803];
  }else{
    [u32:TMP0803] = 0x00000002 << [w:TMP0800] ;
    [u32:TMP0801] = [u32:TMP0801] & (~[u32:TMP0803]);
  }

  [w:TMP0800] = [w:TMP0800] + 1;
}

[u32:GS531] = [u32:TMP0801]; //Set GS531 to the reflected result.

set([b:GB61015]);
set([b:GD62560.b1]); //Read recipe
```

5.7.3对象脚本

基本画面 30002

对象(名称)	数值显示(1 相电流_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常
<pre>//Calculate the decimal point position. if([s16:GD63996]==0) { //EMU4 switch([1-248:s16:400755]) { case -3: my.decimal_point = 3; break; case -2: my.decimal_point = 2; break; case -1: my.decimal_point = 1; break; case 0: my.decimal_point = 0; break; } }else{ //ME110 switch([s16:GD61025[s16:GD62552]]) { case 1: my.decimal_point = 2; break; case 2: my.decimal_point = 1; break; case 3: my.decimal_point = 0;</pre>			

```

        break;

    case 4: $V = $$ * 10;
        my.decimal_point = 0;
        break;

    default: my.decimal_point = 2;
        break;
}
}
redraw_object();

```

对象(名称)	数值显示(2 相电流_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

```

//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400755])
    {
        case -3: my.decimal_point = 3;
            break;

        case -2: my.decimal_point = 2;
            break;

        case -1: my.decimal_point = 1;
            break;

        case 0: my.decimal_point = 0;
            break;

    }

}
else{
    //ME110
    switch([s16:GD61025[s16:GD62552]])
    {
        case 1: my.decimal_point = 2;
            break;

        case 2: my.decimal_point = 1;
            break;

        case 3: my.decimal_point = 0;
            break;

        case 4: $V = $$ * 10;
            my.decimal_point = 0;
            break;

        default: my.decimal_point = 2;
            break;

    }

}
}

```


redraw_object();			
对象(名称)	数值显示(3 相电流_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常
<pre>//Calculate the decimal point position. if([s16:GD63996]==0) { //EMU4 switch([1-248:s16:400755]) { case -3: my.decimal_point = 3; break; case -2: my.decimal_point = 2; break; case -1: my.decimal_point = 1; break; case 0: my.decimal_point = 0; break; } }else{ //ME110 switch([s16:GD61025[s16:GD62552]]) { case 1: my.decimal_point = 2; break; case 2: my.decimal_point = 1; break; case 3: my.decimal_point = 0; break; case 4: \$V = \$\$ * 10; my.decimal_point = 0; break; default: my.decimal_point = 2; break; } } redraw_object();</pre>			
对象(名称)	数值显示(1 相电流需求_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常
<pre>//Calculate the decimal point position. if([s16:GD63996]==0) { //EMU4 switch([1-248:s16:400755])</pre>			

```

{
    case -3: my.decimal_point = 3;
        break;

    case -2: my.decimal_point = 2;
        break;

    case -1: my.decimal_point = 1;
        break;

    case 0:  my.decimal_point = 0;
        break;

}

}else{
    //ME110
    switch([s16:GD61025[s16:GD62552]])
    {
        case 1: my.decimal_point = 2;
            break;

        case 2: my.decimal_point = 1;
            break;

        case 3: my.decimal_point = 0;
            break;

        case 4: $V = $$ * 10;
            my.decimal_point = 0;
            break;

        default: my.decimal_point = 2;
            break;
    }
}

redraw_object();//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400755])
    {
        case -3: my.decimal_point = 3;
            break;

        case -2: my.decimal_point = 2;
            break;

        case -1: my.decimal_point = 1;
            break;

        case 0:  my.decimal_point = 0;
            break;

    }
}

```

```

    }else{
        //ME110
        switch([s16:GD61025[s16:GD62552]])
        {
            case 1: my.decimal_point = 2;
                    break;

            case 2: my.decimal_point = 1;
                    break;

            case 3: my.decimal_point = 0;
                    break;

            case 4: $V = $$ * 10;
                    my.decimal_point = 0;
                    break;

            default: my.decimal_point = 2;
                    break;
        }
    }
}

redraw_object();

```

对象(名称)	数值显示(2 相电流需求_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

```

//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400755])
    {
        case -3: my.decimal_point = 3;
                break;

        case -2: my.decimal_point = 2;
                break;

        case -1: my.decimal_point = 1;
                break;

        case 0:  my.decimal_point = 0;
                break;

    }
}

}else{
    //ME110
    switch([s16:GD61025[s16:GD62552]])
    {
        case 1: my.decimal_point = 2;
                break;

        case 2: my.decimal_point = 1;
                break;
    }
}

```

```

        case 3: my.decimal_point = 0;
                break;

        case 4: $V = $$ * 10;
                my.decimal_point = 0;
                break;

        default: my.decimal_point = 2;
                break;
    }
}

redraw_object();

```

对象(名称)	数值显示(3 相电流需求_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

```

//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400755])
    {
        case -3: my.decimal_point = 3;
                break;

        case -2: my.decimal_point = 2;
                break;

        case -1: my.decimal_point = 1;
                break;

        case 0: my.decimal_point = 0;
                break;

    }
}
}else{
    //ME110
    switch([s16:GD61025[s16:GD62552]])
    {
        case 1: my.decimal_point = 2;
                break;

        case 2: my.decimal_point = 1;
                break;

        case 3: my.decimal_point = 0;
                break;

        case 4: $V = $$ * 10;
                my.decimal_point = 0;
                break;

        default: my.decimal_point = 2;
                break;
    }
}

```

<pre> } redraw_object(); </pre>			
对象(名称)	数值显示(1-2 线电压_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常
<pre> //Calculate the decimal point position. if([s16:GD63996]==0) { //EMU4 switch([1-248:s16:400756]) { case -1: my.decimal_point = 1; break; case 0: my.decimal_point = 0; break; } }else{ //ME110 switch([s16:GD61026[s16:GD62552]]) { case 1:my.decimal_point = 1; break; case 2:my.decimal_point = 0; break; case 3:\$V = \$\$ * 10; my.decimal_point = 0; break; case 4:\$V = \$\$ * 100; my.decimal_point = 0; break; default: \$V = \$\$ * 10; my.decimal_point = 0; break; } } redraw_object(); </pre>			
对象(名称)	数值显示(2-3 线电压_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常
<pre> //Calculate the decimal point position. if([s16:GD63996]==0) { //EMU4 switch([1-248:s16:400756]) { case -1: my.decimal_point = 1; </pre>			

```

        break;

    case 0:    my.decimal_point = 0;
        break;
}

}else{
    //ME110
    switch([s16:GD61026[s16:GD62552]])
    {
        case 1: my.decimal_point = 1;
            break;

        case 2: my.decimal_point = 0;
            break;

        case 3: $V = $$ * 10;
            my.decimal_point = 0;
            break;

        case 4: $V = $$ * 100;
            my.decimal_point = 0;
            break;

        default: $V = $$ * 10;
            my.decimal_point = 0;
            break;
    }
}

redraw_object();

```

对象(名称)	数值显示(3-1 线电压_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

```

//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400756])
    {
        case -1: my.decimal_point = 1;
            break;

        case 0:    my.decimal_point = 0;
            break;

    }

}else{
    //ME110
    switch([s16:GD61026[s16:GD62552]])
    {
        case 1: my.decimal_point = 1;
            break;

        case 2: my.decimal_point = 0;

```

```

        break;

    case 3: $V = $$ * 10;
        my.decimal_point = 0;
        break;

    case 4: $V = $$ * 100;
        my.decimal_point = 0;
        break;

    default: $V = $$ * 10;
        my.decimal_point = 0;
        break;
}

}

redraw_object();

```

对象(名称)	数值显示(功率_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

```

//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400757])
    {
        case -3: my.decimal_point = 3;
            break;

        case -2: my.decimal_point = 2;
            break;

        case -1: my.decimal_point = 1;
            break;

        case 0: my.decimal_point = 0;
            break;

        case 1: $V = $$ * 10;
            my.decimal_point = 0;
            break;
    }
}
else{
    //ME110
    switch([s16:GD61027[s16:GD62552]])
    {
        case 1: my.decimal_point = 4;
            break;

        case 2: my.decimal_point = 3;
            break;

        case 3: my.decimal_point = 2;
            break;
    }
}

```

```
case 4: my.decimal_point = 1;
    break;

case 5: my.decimal_point = 0;
    break;

case 6: $V = $$ * 10;
    my.decimal_point = 0;
    break;

case 7: $V = $$ * 100;
    my.decimal_point = 0;
    break;

default: my.decimal_point = 2;
    break;
}

}

redraw_object();
```

对象(名称)	数值显示(功率需求_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

```
//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400757])
    {
        case -3: my.decimal_point = 3;
            break;

        case -2: my.decimal_point = 2;
            break;

        case -1: my.decimal_point = 1;
            break;

        case 0: my.decimal_point = 0;
            break;

        case 1: $V = $$ * 10;
            my.decimal_point = 0;
            break;
    }
}

}else{
    //ME110
    switch([s16:GD61027[s16:GD62552]])
    {
        case 1: my.decimal_point = 4;
            break;

        case 2: my.decimal_point = 3;
            break;
```



```

case 3: my.decimal_point = 2;
    break;

case 4: my.decimal_point = 1;
    break;

case 5: my.decimal_point = 0;
    break;

case 6: $V = $$ * 10;
    my.decimal_point = 0;
    break;

case 7: $V = $$ * 100;
    my.decimal_point = 0;
    break;

default: my.decimal_point = 2;
    break;
}

}

redraw_object();

```

对象(名称)	数值显示(无功功率 数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

```

//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400757])
    {
        case -3: my.decimal_point = 3;
            break;

        case -2: my.decimal_point = 2;
            break;

        case -1: my.decimal_point = 1;
            break;

        case 0: my.decimal_point = 0;
            break;

        case 1: $V = $$ * 10;
            my.decimal_point = 0;
            break;
    }
}

}else{
    //ME110
    switch([s16:GD61027[s16:GD62552]])
    {
        case 1: my.decimal_point = 4;
            break;
    }
}

```

```

        case 2: my.decimal_point = 3;
            break;

        case 3: my.decimal_point = 2;
            break;

        case 4: my.decimal_point = 1;
            break;

        case 5: my.decimal_point = 0;
            break;

        case 6: $V = $$ * 10;
            my.decimal_point = 0;
            break;

        case 7: $V = $$ * 100;
            my.decimal_point = 0;
            break;

        default: my.decimal_point = 2;
            break;
    }
}

```

redraw_object();

对象(名称)	数值显示(电能_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

//Calculate the decimal point position.

if([s16:GD63996]==0)

```

{
    //EMU4
    switch([1-248:s16:400758])
    {
        case -2: my.decimal_point = 2;
            break;

        case -1: my.decimal_point = 1;
            break;

        case 0: my.decimal_point = 0;
            break;

        case 1: $V = $$ * 10;
            my.decimal_point = 0;
            break;

        case 2: $V = $$ * 100;
            my.decimal_point = 0;
            break;
    }
}

```

}else{

//ME110

switch([s16:GD61028[s16:GD62552]])

```

{
    case 1: my.decimal_point = 2;
        break;

    case 2: my.decimal_point = 1;
        break;

    case 3: my.decimal_point = 0;
        break;

    case 4: $V = $$ * 10;
        my.decimal_point = 0;
        break;

    case 5: $V = $$ * 100;
        my.decimal_point = 0;
        break;

    case 6: $V = $$ * 1000;
        my.decimal_point = 0;
        break;

    default: my.decimal_point = 1;
        break;
}
}

```

redraw_object();

对象(名称)	数值显示(无功电能_数值)		
脚本用户 ID	1		
数据类型	有符号 BIN32	触发类型	通常

//Calculate the decimal point position.

if([s16:GD63996]==0)

```

{
    //EMU4
    switch([1-248:s16:400758])
    {
        case -2: my.decimal_point = 2;
            break;

        case -1: my.decimal_point = 1;
            break;

        case 0: my.decimal_point = 0;
            break;

        case 1: $V = $$ * 10;
            my.decimal_point = 0;
            break;

        case 2: $V = $$ * 100;
            my.decimal_point = 0;
            break;
    }
}

```

}else{

```
//ME110
switch([s16:GD61028[s16:GD62552]])
{
    case 1: my.decimal_point = 2;
            break;

    case 2: my.decimal_point = 1;
            break;

    case 3: my.decimal_point = 0;
            break;

    case 4: $V = $$ * 10;
            my.decimal_point = 0;
            break;

    case 5: $V = $$ * 100;
            my.decimal_point = 0;
            break;

    case 6: $V = $$ * 1000;
            my.decimal_point = 0;
            break;

    default: my.decimal_point = 1;
            break;
}

}

redraw_object();
```

对象(名称)	数值显示(CO2 排出量_数值)		
脚本用户 ID	1		
数据类型	实数	触发类型	通常

```
//Calculate CO2 emission.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400758])
    {
        case -2: $V = $$ * 0.01 * [flt:GD61021[s16:GD62552]];
                my.decimal_point = 2;
                break;

        case -1: $V = $$ * 0.1 * [flt:GD61021[s16:GD62552]];
                my.decimal_point = 1;
                break;

        case 0: $V = $$ * [flt:GD61021[s16:GD62552]];
                break;

        case 1: $V = $$ * 10 * [flt:GD61021[s16:GD62552]];
                break;

        case 2: $V = $$ * 100 * [flt:GD61021[s16:GD62552]];
                break;
    }
}
```

```

}else{
  //ME110
  switch([s16:GD61028[s16:GD62552]])
  {
    case 1: $V = $$ * 0.01 * [flt:GD61021[s16:GD62552]];
      my.decimal_point = 2;
      break;

    case 2: $V = $$ * 0.1 * [flt:GD61021[s16:GD62552]];
      my.decimal_point = 1;
      break;

    case 3: $V = $$ * [flt:GD61021[s16:GD62552]];
      break;

    case 4: $V = $$ * 10 * [flt:GD61021[s16:GD62552]];
      break;

    case 5: $V = $$ * 100 * [flt:GD61021[s16:GD62552]];
      break;

    case 6: $V = $$ * 1000 * [flt:GD61021[s16:GD62552]];
      break;
  }
}
redraw_object();

```

对象(名称)	数值显示(电费_数值)		
脚本用户 ID	1		
数据类型	实数	触发类型	通常

```

//Calculate electric rate.

if([s16:GD63996]==0)
{
  //EMU4
  switch([1-248:s16:400758])
  {
    case -2: $V = $$ * 0.01 * [flt:GD61023[s16:GD62552]];
      break;

    case -1: $V = $$ * 0.1 * [flt:GD61023[s16:GD62552]];
      break;

    case 0: $V = $$ * [flt:GD61023[s16:GD62552]];
      break;

    case 1: $V = $$ * 10 * [flt:GD61023[s16:GD62552]];
      break;

    case 2: $V = $$ * 100 * [flt:GD61023[s16:GD62552]];
      break;
  }
}

}else{
  //ME110

```

```

switch([s16:GD61028[s16:GD62552]])
{
    case 1: $V = $$ * 0.01 * [flt:GD61023[s16:GD62552]];
        break;

    case 2: $V = $$ * 0.1 * [flt:GD61023[s16:GD62552]];
        break;

    case 3: $V = $$ * [flt:GD61023[s16:GD62552]];
        break;

    case 4: $V = $$ * 10 * [flt:GD61023[s16:GD62552]];
        break;

    case 5: $V = $$ * 100 * [flt:GD61023[s16:GD62552]];
        break;

    case 6: $V = $$ * 1000 * [flt:GD61023[s16:GD62552]];
        break;
}

}

redraw_object();

```

基本画面 30009

对象(名称)	数值显示(当前值_No.1)		
脚本用户 ID	1		
数据类型	无符号 BIN32	触发类型	通常
<pre> //Calculate the decimal point position. if([s16:GD63996]==0) { //EMU4 switch([1-248:s16:400758]) { case -2: my.decimal_point = 2; break; case -1: my.decimal_point = 1; break; case 0: my.decimal_point = 0; break; case 1: \$V = \$\$ * 10; my.decimal_point = 0; break; case 2: \$V = \$\$ * 100; my.decimal_point = 0; break; } } else{ //ME110 switch([s16:GD61028[s16:GD62552]]) { </pre>			

```
case 1: my.decimal_point = 2;
    break;

case 2: my.decimal_point = 1;
    break;

case 3: my.decimal_point = 0;
    break;

case 4: $V = $$ * 10;
    my.decimal_point = 0;
    break;

case 5: $V = $$ * 100;
    my.decimal_point = 0;
    break;

case 6: $V = $$ * 1000;
    my.decimal_point = 0;
    break;
}

}

redraw_object();
```

对象(名称)	数值显示(光标信息_No.1)		
脚本用户 ID	1		
数据类型	有符号 BIN16	触发类型	通常

```
//Calculate the decimal point position.
if([s16:GD63996]==0)
{
    //EMU4
    switch([1-248:s16:400758])
    {
        case -2: my.decimal_point = 2;
            break;

        case -1: my.decimal_point = 1;
            break;

        case 0: my.decimal_point = 0;
            break;

        case 1: my.decimal_point = 0;
            break;

        case 2: my.decimal_point = 0;
            break;
    }
}
}else{
    //ME110
    switch([s16:GD61028[s16:GD62552]])
    {
        case 1: my.decimal_point = 2;
            break;
```

```
        case 2: my.decimal_point = 1;
            break;

        case 3: my.decimal_point = 0;
            break;

        case 4: my.decimal_point = 0;
            break;

        case 5: my.decimal_point = 0;
            break;

        case 6: my.decimal_point = 0;
            break;
    }

}

redraw_object();
```

基本画面 30010

对象(名称)	数值显示(当前值_No.1)		
脚本用户 ID	1		
数据类型	无符号 BIN32	触发类型	通常
<pre>//Calculate the decimal point position. if([s16:GD63996]==0) { //EMU4 switch([1-248:s16:400755]) { case -3: my.decimal_point = 3; break; case -2: my.decimal_point = 2; break; case -1: my.decimal_point = 1; break; case 0: my.decimal_point = 0; break; } } else{ //ME110 switch([s16:GD61025[s16:GD62552]]) { case 1: my.decimal_point = 2; break; case 2: my.decimal_point = 1; break; case 3: my.decimal_point = 0; break;</pre>			

<pre> case 4: \$V = \$\$ * 10; my.decimal_point = 0; break; } } redraw_object(); </pre>			
对象(名称)	数值显示(光标信息_No.1)		
脚本用户 ID	1		
数据类型	有符号 BIN16	触发类型	通常
<pre> //Calculate the decimal point position. if([s16:GD63996]==0) { //EMU4 switch([1-248:s16:400755]) { case -3: my.decimal_point = 3; break; case -2: my.decimal_point = 2; break; case -1: my.decimal_point = 1; break; case 0: my.decimal_point = 0; break; } } else{ //ME110 switch([s16:GD61025[s16:GD62552]]) { case 1: my.decimal_point = 2; break; case 2: my.decimal_point = 1; break; case 3: my.decimal_point = 0; break; case 4: my.decimal_point = 0; break; } } redraw_object(); </pre>			

基本画面 30020

对象(名称)	注释显示(字)(相线系统)		
脚本用户 ID	1		
数据类型	无符号 BIN16	触发类型	通常
<pre>//Change displayed comment according to the model. if([w:GD62553] >= 1) { [w:GD62608] = [1-248:w:400513] + 10; }else{ [w:GD62608] = [1-248:w:400513]; }</pre>			

窗口画面 30003

对象(名称)	数值显示(年)		
脚本用户 ID	1		
数据类型	无符号 BIN16	触发类型	上升沿 GB40
<pre>//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</pre>			

[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(名称)	数值显示(月)		
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(名称)	数值显示(日)		
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(名称)	数值显示(时)		
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(名称)	数值显示(分)		
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(名称)	数值显示(秒)		
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;

```

6 关于启动时的系统报警(通讯超时)

立即启动后,因对被用户报警所指定的站号进行访问,所以如果存在没有连接设备的站号,将显示系统报警(通讯超时)。即使将不存在的站号正确地设置为无效站号,因启动时切断通讯对象之前该设备将被进行访问,所以将无法抑制系统报警。

如果使用 GT Designer3 实施以下①处理的话,将可以抑制立即启动后的无需求的系统报警。

①删除访问用户报警监视「30001 报警 1」的不存在站号的软元件设置。

根据下列对应表,请删除未连接站号的该软元件设置。

No.	软元件	站号
1	1-1 400525. b2	1
2	1-1 400525. b3	1
3	1-1 400525. b8	1
4	1-1 400525. b9	1
5	1-1 400525. b10	1
6	1-1 400525. b11	1
7	1-1 400525. b12	1
8	1-1 400525. b13	1
9	1-1 400525. b14	1
10	1-1 400525. b15	1
11	1-2 400525. b2	2
12	1-2 400525. b3	2
13	1-2 400525. b8	2
14	1-2 400525. b9	2
15	1-2 400525. b10	2
16	1-2 400525. b11	2
17	1-2 400525. b12	2
18	1-2 400525. b13	2
19	1-2 400525. b14	2
20	1-2 400525. b15	2
21	1-3 400525. b2	3
22	1-3 400525. b3	3
23	1-3 400525. b8	3
24	1-3 400525. b9	3
25	1-3 400525. b10	3
26	1-3 400525. b11	3
27	1-3 400525. b12	3
28	1-3 400525. b13	3
29	1-3 400525. b14	3
30	1-3 400525. b15	3
31	1-4 400525. b2	4
32	1-4 400525. b3	4
33	1-4 400525. b8	4
34	1-4 400525. b9	4
35	1-4 400525. b10	4
36	1-4 400525. b11	4
37	1-4 400525. b12	4
38	1-4 400525. b13	4
39	1-4 400525. b14	4
40	1-4 400525. b15	4
41	1-5 400525. b2	5
42	1-5 400525. b3	5
43	1-5 400525. b8	5

44	1-5 400525. b9	5
45	1-5 400525. b10	5
46	1-5 400525. b11	5
47	1-5 400525. b12	5
48	1-5 400525. b13	5
49	1-5 400525. b14	5
50	1-5 400525. b15	5
51	1-6 400525. b2	6
52	1-6 400525. b3	6
53	1-6 400525. b8	6
54	1-6 400525. b9	6
55	1-6 400525. b10	6
56	1-6 400525. b11	6
57	1-6 400525. b12	6
58	1-6 400525. b13	6
59	1-6 400525. b14	6
60	1-6 400525. b15	6
61	1-7 400525. b2	7
62	1-7 400525. b3	7
63	1-7 400525. b8	7
64	1-7 400525. b9	7
65	1-7 400525. b10	7
66	1-7 400525. b11	7
67	1-7 400525. b12	7
68	1-7 400525. b13	7
69	1-7 400525. b14	7
70	1-7 400525. b15	7
71	1-8 400525. b2	8
72	1-8 400525. b3	8
73	1-8 400525. b8	8
74	1-8 400525. b9	8
75	1-8 400525. b10	8
76	1-8 400525. b11	8
77	1-8 400525. b12	8
78	1-8 400525. b13	8
79	1-8 400525. b14	8
80	1-8 400525. b15	8
81	1-9 400525. b2	9
82	1-9 400525. b3	9
83	1-9 400525. b8	9
84	1-9 400525. b9	9
85	1-9 400525. b10	9
86	1-9 400525. b11	9
87	1-9 400525. b12	9
88	1-9 400525. b13	9
89	1-9 400525. b14	9
90	1-9 400525. b15	9
91	1-10 400525. b2	10
92	1-10 400525. b3	10
93	1-10 400525. b8	10
94	1-10 400525. b9	10
95	1-10 400525. b10	10

96	1-10 400525. b11	10
97	1-10 400525. b12	10
98	1-10 400525. b13	10
99	1-10 400525. b14	10
100	1-10 400525. b15	10
101	1-11 400525. b2	11
102	1-11 400525. b3	11
103	1-11 400525. b8	11
104	1-11 400525. b9	11
105	1-11 400525. b10	11
106	1-11 400525. b11	11
107	1-11 400525. b12	11
108	1-11 400525. b13	11
109	1-11 400525. b14	11
110	1-11 400525. b15	11
111	1-12 400525. b2	12
112	1-12 400525. b3	12
113	1-12 400525. b8	12
114	1-12 400525. b9	12
115	1-12 400525. b10	12
116	1-12 400525. b11	12
117	1-12 400525. b12	12
118	1-12 400525. b13	12
119	1-12 400525. b14	12
120	1-12 400525. b15	12
121	1-13 400525. b2	13
122	1-13 400525. b3	13
123	1-13 400525. b8	13
124	1-13 400525. b9	13
125	1-13 400525. b10	13
126	1-13 400525. b11	13
127	1-13 400525. b12	13
128	1-13 400525. b13	13
129	1-13 400525. b14	13
130	1-13 400525. b15	13
131	1-14 400525. b2	14
132	1-14 400525. b3	14
133	1-14 400525. b8	14
134	1-14 400525. b9	14
135	1-14 400525. b10	14
136	1-14 400525. b11	14
137	1-14 400525. b12	14
138	1-14 400525. b13	14
139	1-14 400525. b14	14
140	1-14 400525. b15	14
141	1-15 400525. b2	15
142	1-15 400525. b3	15
143	1-15 400525. b8	15
144	1-15 400525. b9	15
145	1-15 400525. b10	15
146	1-15 400525. b11	15
147	1-15 400525. b12	15

148	1-15 400525. b13	15
149	1-15 400525. b14	15
150	1-15 400525. b15	15
151	1-16 400525. b2	16
152	1-16 400525. b3	16
153	1-16 400525. b8	16
154	1-16 400525. b9	16
155	1-16 400525. b10	16
156	1-16 400525. b11	16
157	1-16 400525. b12	16
158	1-16 400525. b13	16
159	1-16 400525. b14	16
160	1-16 400525. b15	16
161	1-17 400525. b2	17
162	1-17 400525. b3	17
163	1-17 400525. b8	17
164	1-17 400525. b9	17
165	1-17 400525. b10	17
166	1-17 400525. b11	17
167	1-17 400525. b12	17
168	1-17 400525. b13	17
169	1-17 400525. b14	17
170	1-17 400525. b15	17
171	1-18 400525. b2	18
172	1-18 400525. b3	18
173	1-18 400525. b8	18
174	1-18 400525. b9	18
175	1-18 400525. b10	18
176	1-18 400525. b11	18
177	1-18 400525. b12	18
178	1-18 400525. b13	18
179	1-18 400525. b14	18
180	1-18 400525. b15	18
181	1-19 400525. b2	19
182	1-19 400525. b3	19
183	1-19 400525. b8	19
184	1-19 400525. b9	19
185	1-19 400525. b10	19
186	1-19 400525. b11	19
187	1-19 400525. b12	19
188	1-19 400525. b13	19
189	1-19 400525. b14	19
190	1-19 400525. b15	19
191	1-20 400525. b2	20
192	1-20 400525. b3	20
193	1-20 400525. b8	20
194	1-20 400525. b9	20
195	1-20 400525. b10	20
196	1-20 400525. b11	20
197	1-20 400525. b12	20
198	1-20 400525. b13	20
199	1-20 400525. b14	20

200	1-20 400525. b15	20
201	1-21 400525. b2	21
202	1-21 400525. b3	21
203	1-21 400525. b8	21
204	1-21 400525. b9	21
205	1-21 400525. b10	21
206	1-21 400525. b11	21
207	1-21 400525. b12	21
208	1-21 400525. b13	21
209	1-21 400525. b14	21
210	1-21 400525. b15	21
211	1-22 400525. b2	22
212	1-22 400525. b3	22
213	1-22 400525. b8	22
214	1-22 400525. b9	22
215	1-22 400525. b10	22
216	1-22 400525. b11	22
217	1-22 400525. b12	22
218	1-22 400525. b13	22
219	1-22 400525. b14	22
220	1-22 400525. b15	22
221	1-23 400525. b2	23
222	1-23 400525. b3	23
223	1-23 400525. b8	23
224	1-23 400525. b9	23
225	1-23 400525. b10	23
226	1-23 400525. b11	23
227	1-23 400525. b12	23
228	1-23 400525. b13	23
229	1-23 400525. b14	23
230	1-23 400525. b15	23
231	1-24 400525. b2	24
232	1-24 400525. b3	24
233	1-24 400525. b8	24
234	1-24 400525. b9	24
235	1-24 400525. b10	24
236	1-24 400525. b11	24
237	1-24 400525. b12	24
238	1-24 400525. b13	24
239	1-24 400525. b14	24
240	1-24 400525. b15	24
241	1-25 400525. b2	25
242	1-25 400525. b3	25
243	1-25 400525. b8	25
244	1-25 400525. b9	25
245	1-25 400525. b10	25
246	1-25 400525. b11	25
247	1-25 400525. b12	25
248	1-25 400525. b13	25
249	1-25 400525. b14	25
250	1-25 400525. b15	25
251	1-26 400525. b2	26

252	1-26 400525. b3	26
253	1-26 400525. b8	26
254	1-26 400525. b9	26
255	1-26 400525. b10	26
256	1-26 400525. b11	26
257	1-26 400525. b12	26
258	1-26 400525. b13	26
259	1-26 400525. b14	26
260	1-26 400525. b15	26
261	1-27 400525. b2	27
262	1-27 400525. b3	27
263	1-27 400525. b8	27
264	1-27 400525. b9	27
265	1-27 400525. b10	27
266	1-27 400525. b11	27
267	1-27 400525. b12	27
268	1-27 400525. b13	27
269	1-27 400525. b14	27
270	1-27 400525. b15	27
271	1-28 400525. b2	28
272	1-28 400525. b3	28
273	1-28 400525. b8	28
274	1-28 400525. b9	28
275	1-28 400525. b10	28
276	1-28 400525. b11	28
277	1-28 400525. b12	28
278	1-28 400525. b13	28
279	1-28 400525. b14	28
280	1-28 400525. b15	28
281	1-29 400525. b2	29
282	1-29 400525. b3	29
283	1-29 400525. b8	29
284	1-29 400525. b9	29
285	1-29 400525. b10	29
286	1-29 400525. b11	29
287	1-29 400525. b12	29
288	1-29 400525. b13	29
289	1-29 400525. b14	29
290	1-29 400525. b15	29
291	1-30 400525. b2	30
292	1-30 400525. b3	30
293	1-30 400525. b8	30
294	1-30 400525. b9	30
295	1-30 400525. b10	30
296	1-30 400525. b11	30
297	1-30 400525. b12	30
298	1-30 400525. b13	30
299	1-30 400525. b14	30
300	1-30 400525. b15	30
301	1-31 400525. b2	31
302	1-31 400525. b3	31
303	1-31 400525. b8	31

304	1-31 400525. b9	31
305	1-31 400525. b10	31
306	1-31 400525. b11	31
307	1-31 400525. b12	31
308	1-31 400525. b13	31
309	1-31 400525. b14	31
310	1-31 400525. b15	31