



HMI 产品

Pro-face

GP 系列

可编程、触摸式工业图形显示器，GP系列

高级应用手册

上海天任电子有限公司

2003 年 04 月

前言

承蒙使用日本 Digital 公司 Pro-face GP 系列触摸屏工业图形显示器产品，万分感谢。

如需要了解：

GP 的硬件，参考：GP 用户手册。

一般的应用基础，参考：GP 操作应用基础、GP 操作指导。

高级的应用指导，参考：高级应用(本手册)、Tag 使用手册。

本手册为 GP-PRO/PBIII 软件的高级功能应用指导，并不包含一般应用基础知识。

本书涉及的例子，都将可以在本公司网站的下载中心下载。但是由于软件更新，可能会与这里所述有所区别。

本书涉及的例子，都可能以某种特定的 PLC 类型为例。下载后，可以在 GP-PRO/PBIII 软件里改变（但是注意确认所有地址的正确性）；或者使用 GP-PRO/PBIII 的 Simulate 功能。

本书的说明，以 GP-PRO/PBIII V3.0 以上版本为准，建议使用以 GP-PRO/PBIII V5.05 以上版本。特殊注明的需要更高版本。

在使用中，如还有需了解的信息，请与我们联系。

上海天任电子有限公司作为 DIGITAL 在中国唯一的 GP 产品技术服务中心，愿竭诚为您服务。

网址：<http://www.tianren88.com>

技术：support@tianren88.com

目录

1	变量(SYMBOL)定义及应用.....	1-1
1.1	变量(SYMBOL)定义的优点	1-1
1.2	变量(SYMBOL)定义.....	1-1
1.3	变量(SYMBOL)的使用	1-3
2	D 脚本编程语言 (D-SCRIPT)	2-1
2.1	选择 SPECIAL(特殊)下拉菜单.....	2-1
2.2	选择 D-SCRIPT (D 脚本) 弹出 D-SCRIPT(D 脚本列表).....	2-1
2.3	ADD(添加)按钮, 编辑增加 D-SCRIPT(D 脚本对话框).....	2-2
2.4	OPTIONS(设置)下拉菜单	2-2
2.5	TRIGGER (触发栏)	2-3
2.5.1	定时周期触发.....	2-3
2.5.2	位边沿触发.....	2-3
2.5.3	表达式触发.....	2-3
2.6	D-SCRIPT TOOLBOX(D 脚本工具箱).....	2-4
2.6.1	Functions(功能) 下拉列表框.....	2-4
2.6.2	Statements(编程语句):	2-12
2.6.3	Operators(运算操作符).....	2-13
2.6.4	Compare(比较命令).....	2-15
2.7	功能块.....	2-15
3	弹出键盘窗口	3-1
3.1	弹出键盘的应用	3-1
3.2	自动弹出键盘(v60 以后)	3-1
3.3	U-TAG 实现弹出键盘的制作过程	3-1
3.4	窗口使用的限制.....	3-5
4	密码制作	4-1
4.1	一级密码制作的效果	4-1
4.2	一级密码制作的具体介绍	4-1
4.3	多级密码制作	4-5
5	多语言在线切换	5-1
5.1	多语言的应用	5-1
5.2	多语言的字符串表	5-2
5.2.1	多语言表的控制地址.....	5-2
5.2.2	表的语言类型、名称.....	5-2
5.2.3	初始语言表的设置.....	5-3
5.3	编辑画面时使用语言表	5-4
5.4	运行画面时改变语言表	5-5
5.5	多语言表的限制.....	5-5
6	CF 卡的使用	6-1
6.1	CF 卡的特性.....	6-1

6.2	CF 卡使用注意	6-1
6.2.1	适配器是必须的.....	6-1
6.2.2	存放数据时需预先注意.....	6-1
6.2.3	存取 CF 卡时需注意的事项.....	6-1
6.2.4	例子.....	6-2
6.2.5	其他注意事项.....	6-2
6.3	CF 卡数据输出文件夹设置.....	6-3
6.4	输出文件夹传数据到 CF 卡.....	6-4
6.5	选用 GP 的离线方式.....	6-4
6.6	传送后备 SRAM 中的数据到 CF 卡	6-5
6.6.1.	定义一个字地址.....	6-5
6.6.2.	传送数据到 CF 卡.....	6-5
6.6.3.	模式区.....	6-5
6.6.4.	文件编号.....	6-6
6.6.5.	记录数据循环自动保存.....	6-6
6.7	画面获取.....	6-7
6.7.1.	控制字.....	6-7
6.7.2.	画面获取的设置.....	6-9
6.7.3.	自动增加文件编号功能.....	6-9
6.7.4.	黑白反相.....	6-9
6.8	检查 CF 卡的可用容量	6-10
7	数据记录功能	7-1
7.1	数据记录的应用	7-1
7.2	数据记录占用 SRAM 的计算.....	7-2
7.3	数据记录的设定.....	7-3
7.3.1	触发方式的数据记录设定、记录时序.....	7-3
7.3.2	定时方式的数据记录设定、记录时序.....	7-4
	数据记录的显示设定.....	7-6
	数据记录的打印设定.....	7-7
7.3.5	数据记录的 LS 传送设定.....	7-8
7.3.6	数据记录在画面上的显示设定.....	7-10
8	配方功能.....	8-1
8.1	概述.....	8-1
8.2	配方功能.....	8-1
8.3	文档数据设置.....	8-8
8.4	文档数据列表.....	8-11
8.4.1	文档数据列表.....	8-11
8.4.2	文档数据注册.....	8-12
8.5	文档数据设定	8-14
8.6	文档数据自动传输	8-15
8.7	手动文档数据传输实例 1	8-17
8.8	手动文档数据传输实例 2	8-21

9	扩展串口的应用	9-1
9.1	接收功能.....	9-1
9.2	发送功能.....	9-1
9.3	控制.....	9-2
9.4	状态.....	9-3
9.5	接收数据的大小	9-4
9.6	扩展通讯口的设置	9-4
9.7	通讯举例.....	9-4
10	MEMORY-LINK 通讯.....	10-1
10.1	关于 MEMORY-LINK.....	10-1
10.2	通讯基础.....	10-1
10.3	MEMORY-LINK 通讯基础	10-2
10.3.1	Memory-link 通讯时 GP 内部数据区	10-2
10.3.2	Memory-link 通讯时 GP 内部系统数据区.....	10-3
10.3.3	Memory-link 通讯时电缆连接.....	10-4
10.4	MEMORY-LINK 通讯协议	10-5
10.4.1	ASCII 码兼容方式(Compatible)通讯命令	10-5
10.4.2	ASCII 码兼容方式(Compatible)通讯举例.....	10-6
10.4.3	扩展方式(Extended)通讯命令.....	10-6
11	报警显示.....	11-1
11.1	A-TAG 当前报警显示.....	11-1
11.1.1	关于 A-TAG	11-1
11.1.2	使用 A-tag	11-2
11.2	Q-TAG 警报摘要显示.....	11-10
11.2.1	关于 Q-tag	11-10
11.2.2	Q-tag 的使用限制.....	11-11
11.2.3	使用 Q-Tag	11-11
12	声音输出	12-1
12.1	概述	12-1
12.2	声音输出设置.....	12-2
12.3	声音输出设置范例。	12-3
12.3.1	创建音频文件。	12-3
12.3.2	修改音频文件.....	12-4
12.3.3	删除音频文件.....	12-4

1 变量(symbol)定义及应用

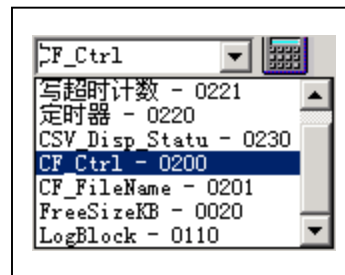
1.1 变量(symbol)定义的优点

在 GP-PRO/PBIII 的画面编辑时，无疑要指定大量的 PLC 功能存储器或 GP 内部 LS 区的地址。GP-PRO/PBIII 的方便之处在于：对 PLC 或 GP 内部的存储器地址，可以随时使用，不需要先定义变量对应。因此调用 Parts 功能非常简单，只需要选择一种部件，然后指定一个地址，即可下载到 GP 上运行。

GP-PRO/PBIII 也支持变量(symbol)定义的方式。即先定义一个变量名称，指定一个对应的 PLC 或 GP 内部存储器地址，然后在画面编辑时直接使用这个变量名称。

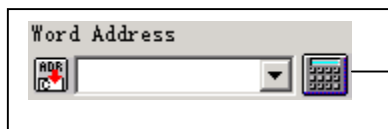
这样，带来以下一些优点：

- 变量名称对应存储器地址，使用时通俗易懂。例如右图中：“定时器 - 0220”。
- 当需要输入地址时，可以直接在下拉列表框里选择，这样既省了重复输入的麻烦，也保证了输入的正确性。
- 当整个工程中需要进行地址变换时，无需一个个地对整个工程里的所有用到该地址的地方去修改。虽然软件也提供了地址变换功能，但是通过变量(symbol)定义的方式，则更明了。
- 在变量定义表格里，可以了解存储器的使用情况。当然软件也另外提供了地址和变量引用列表功能。
- 可以进行变量的输入、输出操作，支持 CSV、LBE 格式的文件，方便各相关软件的数据共享。
- 变量可以随时定义。

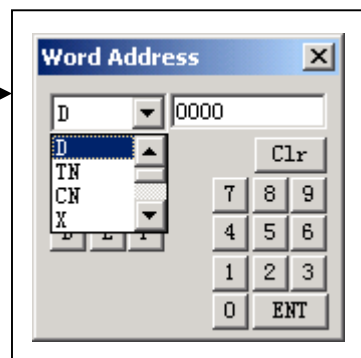


1.2 变量(symbol)定义

当 GP-PRO/PBIII 要求输入存储器地址时，如下图所示，都有一个下拉编辑框，旁边有一个计算器样的小图标。



一般来说，如果不知道地址输入格式的话，应该通过点击该小图标来选择存储器类型并输入地址。如果地址输入超过范围，它是不允许输入的；并且对有效的地址它会自动规格化成标准格式，例如：MITSUBISHI FX2N 的数据寄存器 D，它自动规格化为“Dxxxx”。

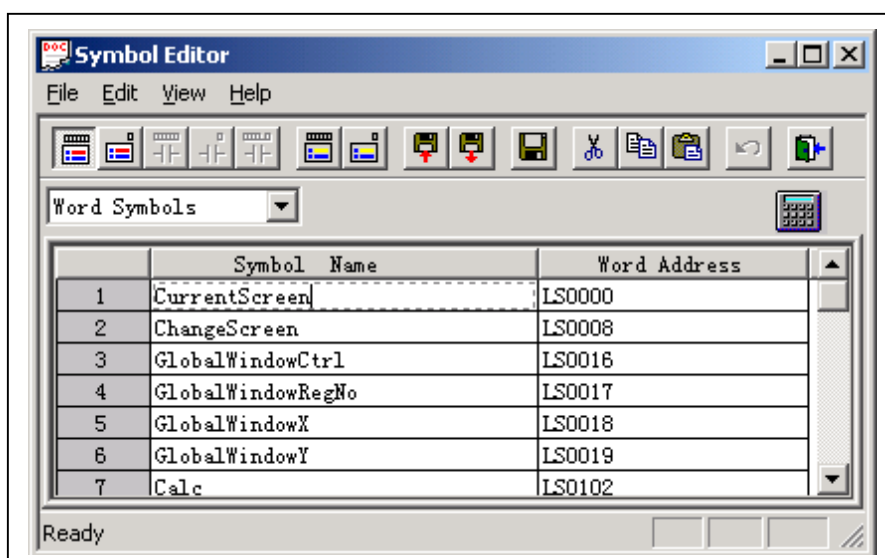


但是，如果是在编辑框里直接输入的不能成为这样的规格化的格式，就会自动出现右边的提示：“是否注册为一个变量？”。这就是变量随时都可以定义的方法。

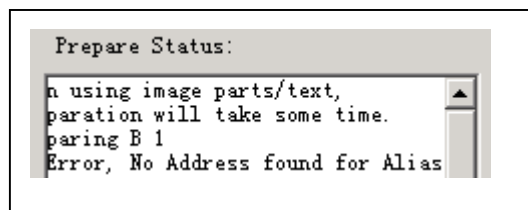


用这种方法定义变量后，就可以随时使用。但是在传送到 GP 之前，还需要到“Symbol Editor”里将变量对应到实际的存储器地址。如下图所示。

比较好的建议是：可以在做画面前，直接在“Symbol Editor”里先定义变量，并将变量对应到实际的存储器地址。



同样需要注意，这里输入的地址则必须规范。如果不做对应、或者对应地址不规范，则在传送到 GP 时会有错误提示。



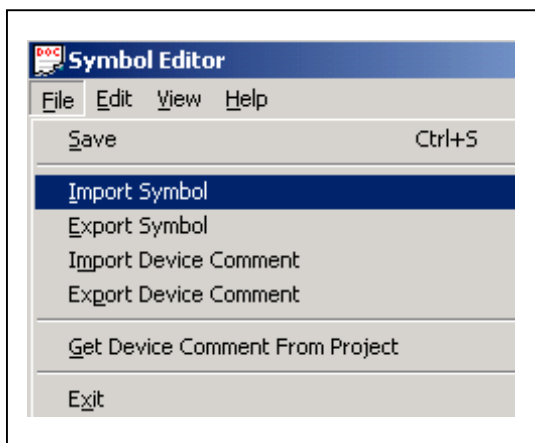
1.3 变量(symbol)的使用

定义的变量分为 Word Address 和 Bit Address。

大部分需要输入地址的地方都可以用定义好的变量(symbol)。但也有极少数例外，有些地方不能直接应用定义好的变量(symbol)。

通常，数据寄存器（“Word Address”）的任意 bit 是可以直接当作 Bit Address 引用的，例如 LS203800，表示的是“Word Address” LS2038 的 bit00。但是，定义好的“Word Address”类型的变量，目前其 bit 不能直接当作“Bit Address”引用。

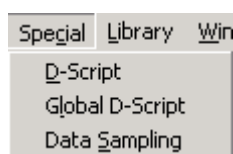
定义的变量可以输入、输出为 CSV、LBE 格式的文件。



2 D 脚本编程语言（D-Script）

在用 GP 软件制作画面的过程之中，根据实际情况，有时候需要对一些数据进行处理，包括判断、运算、转移、复位，甚至包括一些动画的绘制，如果将这些数据全部交给 PLC 处理，必然大大增加 PLC 的负担，减缓通讯速率，并且给 PLC 编程人员加大了编程的难度，为此，GP 提供了一项特殊功能：D 脚本编程语言（D-Script）。使用 D 脚本编写程序，使 GP 本身能够完成一些数据处理，从而协调好与 PLC 的通讯工作。

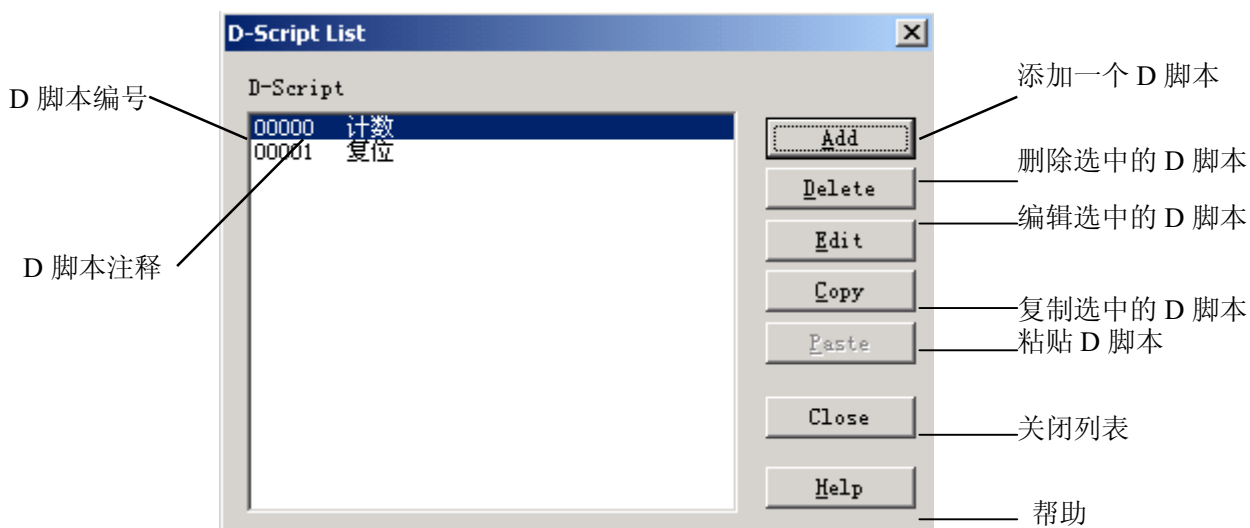
2.1 选择 special(特殊)下拉菜单



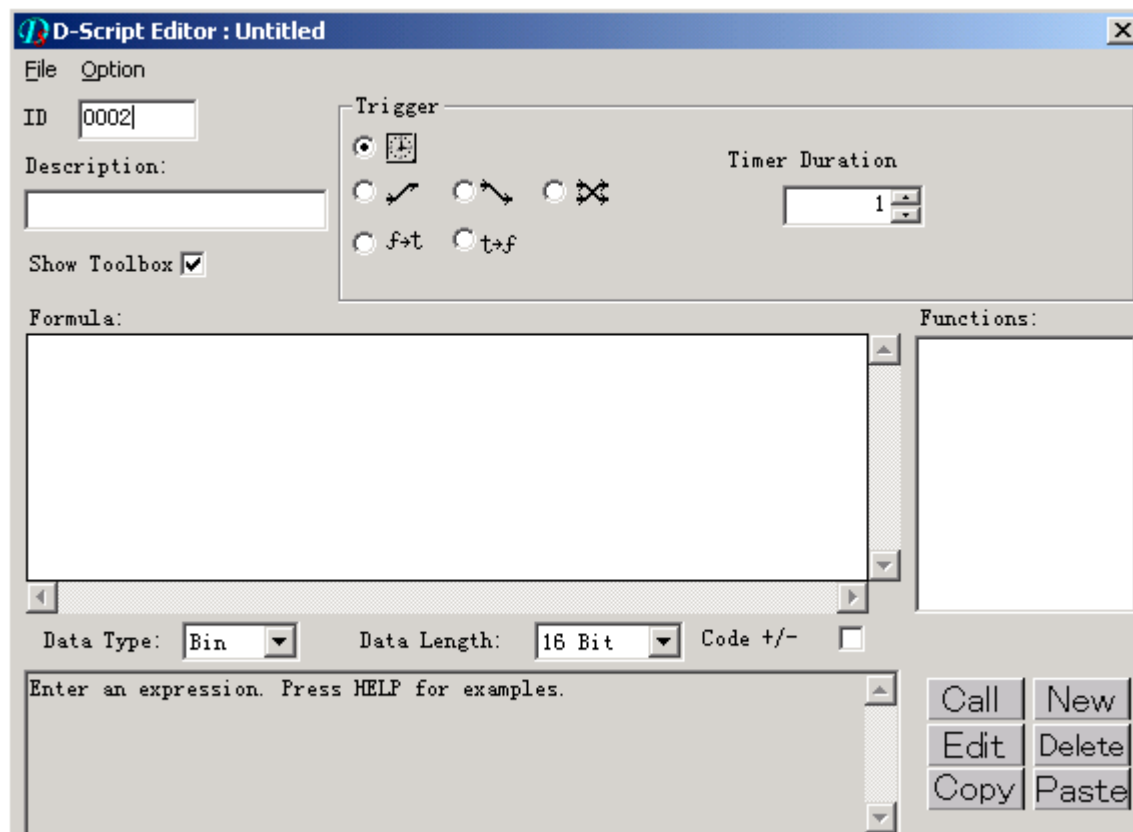
D-Script（D 脚本）仅仅针对当前画面有效。

Global D-Script（全局 D 脚本）对所有画面均有效，适用于 GP77 和 GP2000 系列。

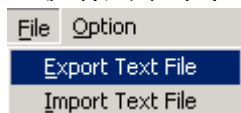
2.2 选择 D-Script（D 脚本）弹出 D-Script(D 脚本列表)



2.3 Add(添加)按钮，编辑增加 D-Script(D 脚本对话框)



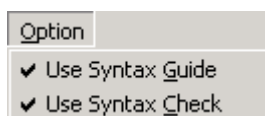
File(文件)下拉菜单:



Export Text File(导出文本文件): 将公式编辑栏里的内容转化为 txt 文本文件后保存到指定目录。

Import Text File(导入文本文件): 将指定目录中的 txt 文本文件内容打开在公式编辑栏里。

2.4 Options(设置)下拉菜单



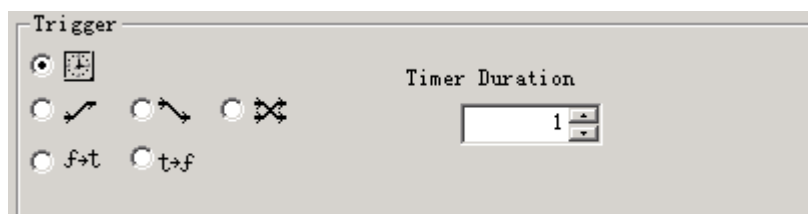
Use Syntax Guide(语法导向): 可以抽取公式编辑栏中的数据。

Use Syntax Guide(语法检查): 自动检查公式编辑栏中的语句是否合法。

2.5 Trigger（触发栏）

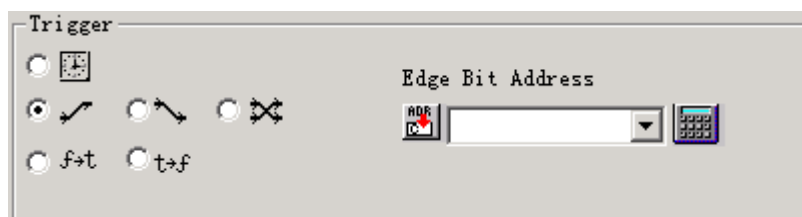
触发栏用于设置启动程序的触发条件，当触发条件成立，便执行公式编辑栏中的程序。触发条件有三种形式：定时周期触发、位边沿触发和表达式条件触发。




2.5.1 定时周期触发



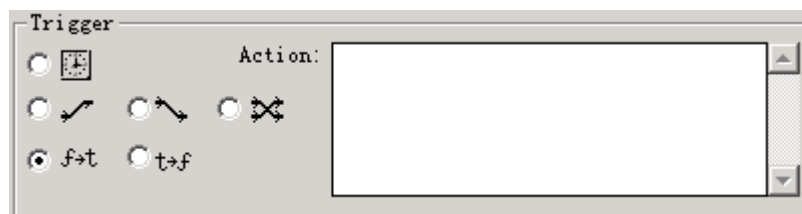
选择定时周期触发时，设置触发时间（秒），则每隔这一段时间，程序执行一次。

2.5.2 位边沿触发



- ：在设定位地址的上升沿触发。
- ：在设定位地址的下降沿触发。
- ：在设定位地址的上升沿和下降沿均触发。

2.5.3 表达式触发



- f→t**：当 Action(操作)栏中的表达式为由假（0）变真（1）时，程序执行一次。
- t→f**：当 Action(操作)栏中的表达式为由真（1）变假（0）时，程序执行一次。

2.6 D-Script Toolbox(D 脚本工具箱)

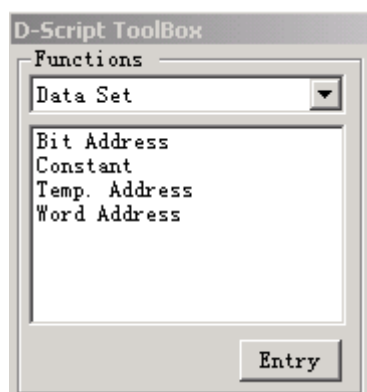
当 Show Toolbox 被选中时,弹出右图所示 D 脚本工具箱,在此工具箱中包含了 D 脚本所用到的所有指令。



2.6.1 Functions(功能)下拉列表框

在 Functions 下拉菜单中有五大部分: Data Set(数据设置)、Draw(绘图)、Memory Ops. (内存操作)、SIO Port Ops. (SIO 口操作)、Bit Ops. (位操作)

2.6.1.1 Data Set(数据设置)



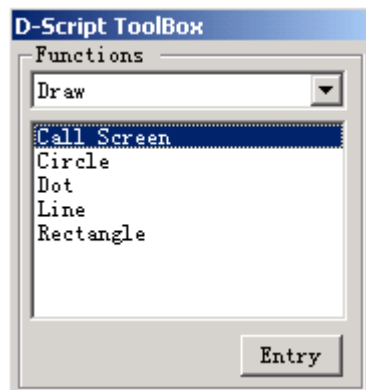
Bit Address:设置位地址。

Constant: 设置常量,有 Dec(十进制)、Oct(八进制)、Hex(十六进制)三种数据格式。

Temp.Address: 设置临时地址

Word Address: 字地址

2.6.1.2 Draw(绘图)



Call Screen: 调用一个画面，需要设置画面号以及目标画面中心所放置的坐标。

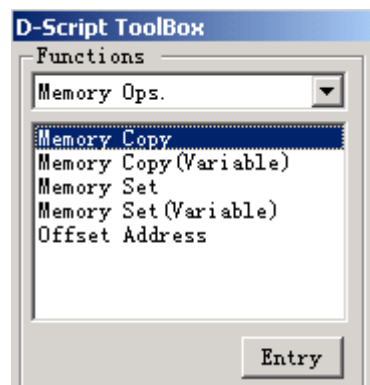
Circle: 绘制圆，需要设置圆中心坐标以及半径。

Dot: 绘制点，需要设置点放置的坐标。

Line: 绘制线段，需要设置线段放置的起始以及终结坐标。

Rectangle: 绘制矩形，需要设置矩形的左上角和右下角坐标。

2.6.1.3 Memory Ops.(内存操作)



Memory Copy (内存复制): 该功能可以将字地址区段复制。

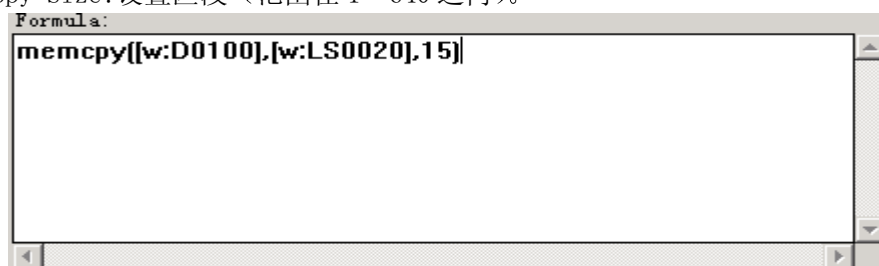
格式: `memcpy([w:DI Word Address], [w:SI Word Address], Copy Size)`

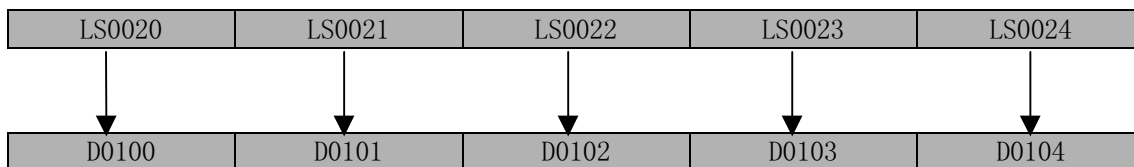
DI Word Address: 内存复制的目标起始字地址。

SI Word Address: 内存复制的源起始字地址。

Copy Size: 设置区段 (范围在 1~640 之内)。

范例:





Memory Copy (Variable): 间接可更改的内存复制。

格式: `_memcpy_EX(Parameter1, Parameter2, Parameter3)`

Parameter1: 带偏移量的目标起始字地址, 偏移量在临时字地址中设置。

Parameter2: 带偏移量的源起始字地址, 偏移量在临时字地址中设置。

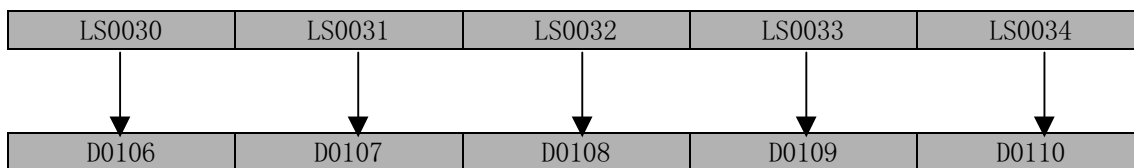
Parameter3: 设置区段, 可以直接用常数给定, 也可以用 LS 字地址或者临时字地址设置 (范围在 1~640 之内)。

范例 :

:

Formula:

```
[t:0000]=6
[t:0001]=10
_memcpy_EX([w:D0100]#[t:0000], [w:LS0020]#[t:0001], 5)
```



Memory Set (内存设定): 可以将字地址区间赋值。

Set Word Address: D0000

Set Data: 0

Set Size: 1

Set Word Address: 设置起始字地址。

Set Data: 给字地址赋值。

Set Size: 设置区段 (范围在 1~640 之内)。

范例

Formula:

```
memset([w:LS0020], 8, 5)
```

该语句将字地址 LS0020~LS0024 赋常量 8。

Memory Set(Variable):间接可更改的字地址区间赋值(仅使用于 GP2000 系列, GP2301 和 GP2501 除外)。

Parameter1:目标起始字地址, 在临时字地址中设定偏移量。

Parameter2:赋值。可以在 LS 区或者临时字地址设置, 或者直接输入常量设置。

Parameter3:设置区段, 可以在 LS 区或者临时字地址设置, 或者直接输入常量设置(范围在 1~640 之内)。

范例

该语句执行下列功能：将常量 80 赋予字地址 D0108~D0113。

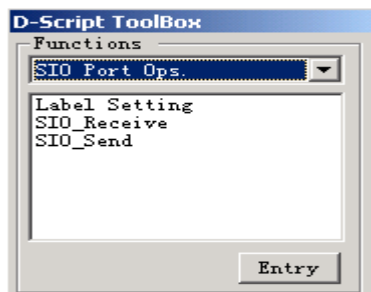
Offset Address:字地址偏移, 该功能适用于 GP77R 系列、377 系列和 GP2000 系列。

格式: [Word Address]#[Temporary Work], 该语句表示一个新的字地址, 即旧地址 Word Address+ Temporary Work 的数值而生成的新地址。

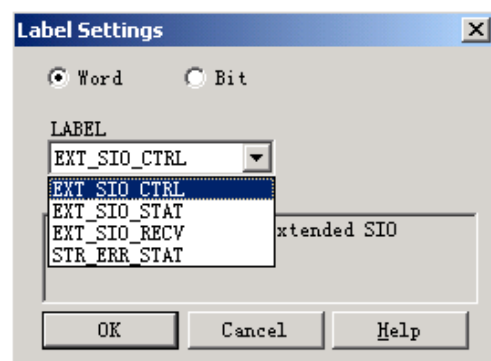
范例

该语句表示: 将字地址 LS0105 的内容赋给字地址 LS0200。

2.6.1.4 SIO port Ops.(扩展串口, 适用 GP2000 系列)



Label Setting



该功能可写, 用于清空输出、清空输入、复位错误状态。

格式: 以“位”为对象时: [c:EXT SIO CTRL**] **代表 00~15

以“字”为对象时: [c:EXT SIO CTRL]

EXT SIO CTRL (扩展口清空控制)

Bit(位)	Constant(内容)
15	保留
14	
13	
12	
11	
10	
9	
8	
7	
6	1: 复位错误
5	
4	
3	
2	
1	
0	

注意: 当 EXT SIO CTRL 被设定后, 程序将以如下顺序执行: 复位错误→清空输入→清空输出

EXT SIO STAT(扩展口状态)

该功能只读，包括如下一些内容。

Bit(位)	Constant(内容)
15	0: 扩展口无通讯协议 1: 扩展口有通讯协议
14	保留
13	
12	
11	
10	
9	
8	
7	
6	
5	
4	
3	0: 正常工作 1: 数据接收错误
2	0: 未接收数据 1: 正接收数据
1	0: 正常工作 1: 传输数据错误
0	0: 输出口有数据 1: 输出口无数据

注意：保留位留待以后设定，勿占用。

数据传输错误有两种：传输间歇和数据堵塞。当两种错误同时存在的时候，传输错误位 ON。

数据接受错误有四种：奇偶错误、满载错误、帧错误、溢出错误。只要其中之一发生，数据接收错误位 ON。

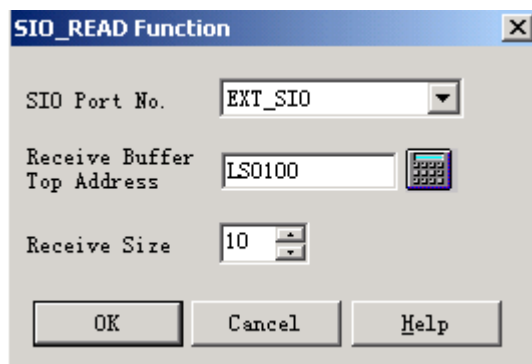
EXT SIO RECV(扩展口输入)

该功能只读，显示接收到的字节数。

格式：[r:EXT_SIO_RECV]

STR ERR STAT(字符串错误状态)

SIO Receive(扩展口数据接收)



从扩展口读入数据写入指定地址寄存器。

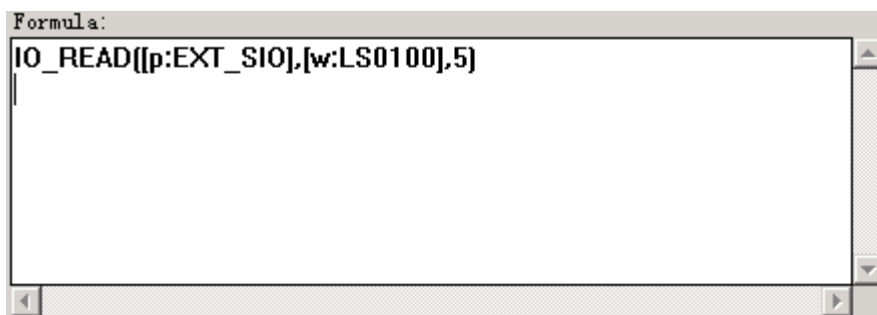
格式: IO_READ([p:SIO Port No.], [w:Receive Buffer Top Address], 10)

SIO Port No. (扩展口): 指定接收数据的扩展口。

Receive Buffer Top Address(数据接收始地址): 起始地址存储接收数据的数量, 其后连续地址存放数据。

Receive Size(数据大小): 设定接收数据的数量。

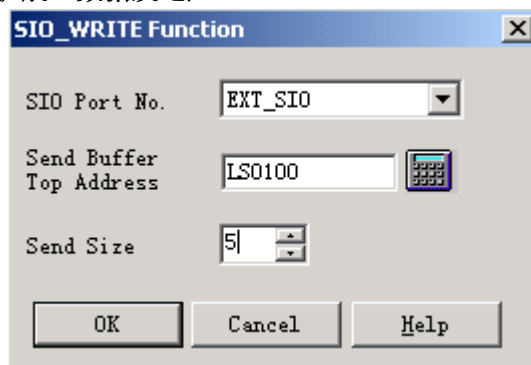
范例



该程序执行:

接收地址	内容
LS0100	5
LS0101	数据 1
LS0102	数据 2
LS0103	数据 3
LS0104	数据 4
LS0105	数据 5

SIO Send(扩展口数据发送)



从指定寄存器写入数据到扩展口
然后由扩展口发送

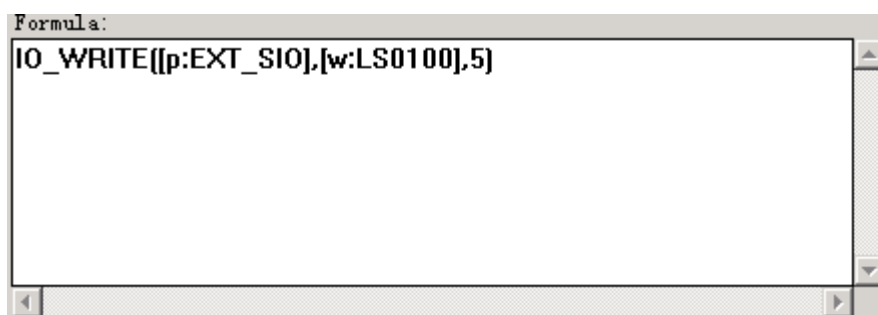
格式: IO_WRITE([p: SIO Port No.],[w:Send Buffer Top Address],Send Size)

SIO Port No. (扩展口):指定发送数据的扩展口。

Send Buffer Top Address(数据发送始地址):指定发送数据的起始源地址。

Send Size(数据大小):指定发送数据的数量。

范例



该程序执行:

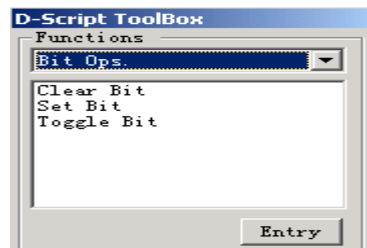
发送地址	内容
LS0100	数据 1
LS0101	数据 2
LS0102	数据 3
LS0103	数据 4
LS0104	数据 5

2.6.1.5Bit Ops.(位操作)

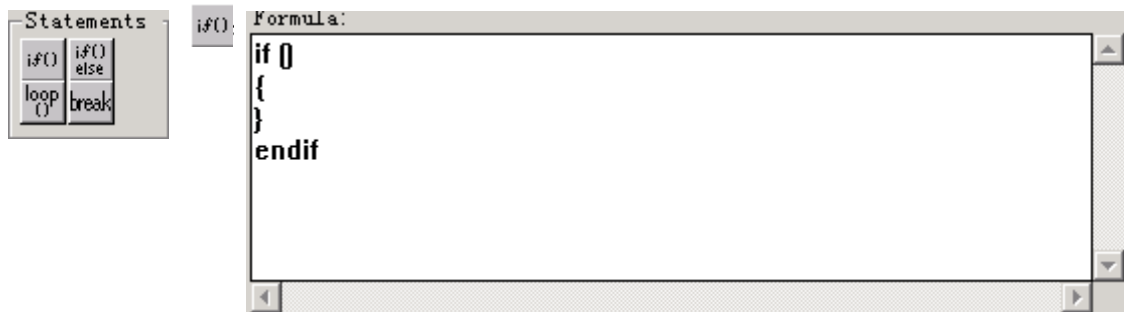
Clear Bit:将指定的位置“0”。

Set Bit:将指定的位置“1”。

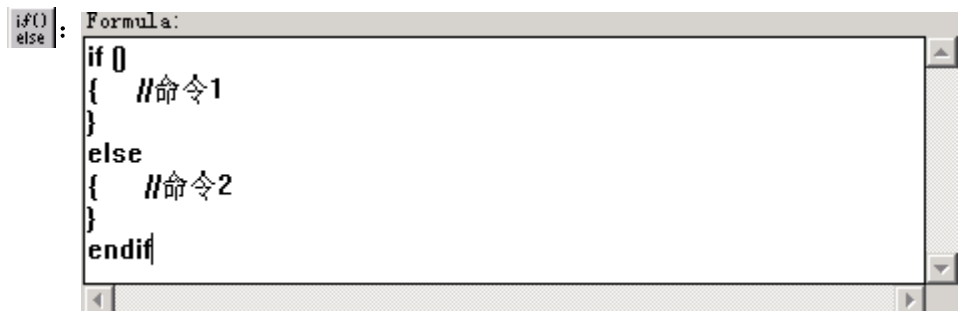
Toggle Bit:将指定的位状态转换, 0→1 或者 1→0。



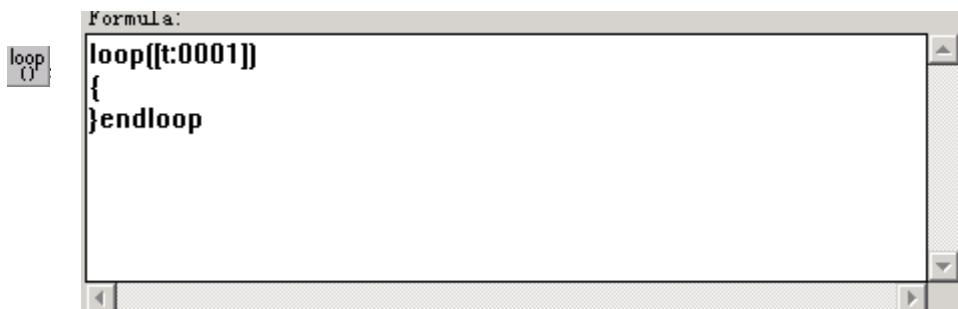
2.6.2 Statements(编程语句):



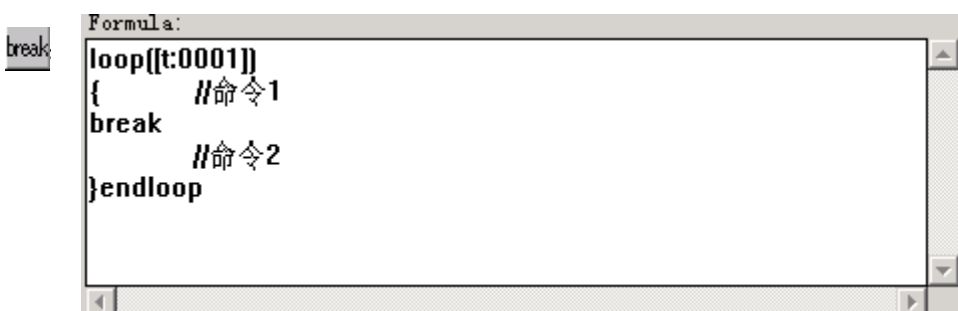
当 () 中的条件为真, 则执行 {} 中的命令。



当 () 中的条件为真, 则执行命令 1, 否则执行命令 2。

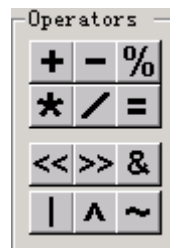


循环执行 {} 中的命令, 循环次数设定于临时地址中。



在执行完命令 1 之后, 不执行命令 2, 结束循环。 Break 指令适用于 GP77R、GP377 和 GP2000 系列。

2.6.3 Operators(运算操作符)

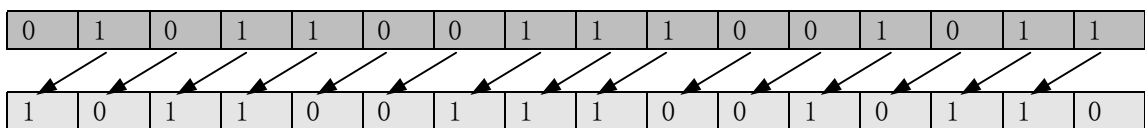
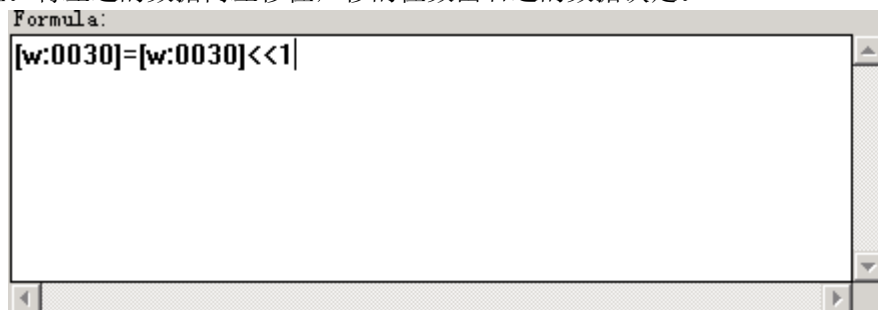


+ 加法
***** 乘法
% 求余

- 减法
/ 除法
= 赋值

<<: 左移位。将左边的数据向左移位，移的位数由右边的数据决定。

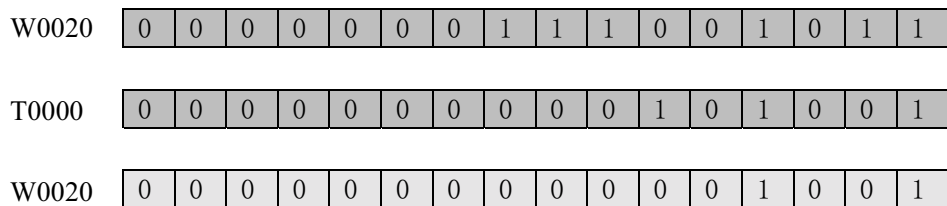
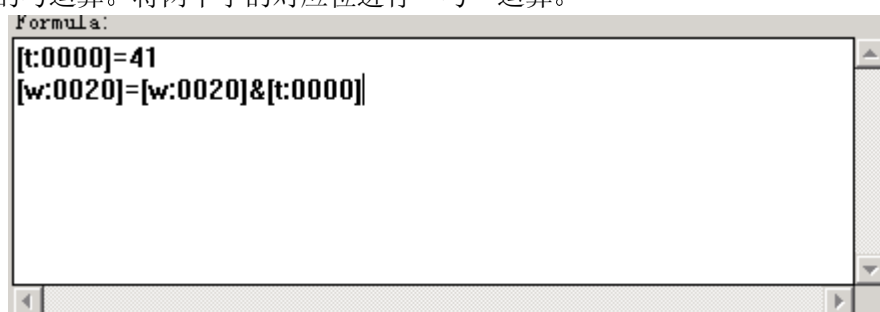
范例:



>>: 右移位。将左边的数据向右移位，移的位数由右边的数据决定。

&: 字的与运算。将两个字的对应位进行“与”运算。

范例:



| : 字的或运算。将两个字的对应位进行“或”运算。

范例

Formula:

```
[t:0000]=41
[w:0020]=[w:0020]|([t:0000])
```

W0020

T0000

0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

W0020

0	0	0	0	0	0	0	1	1	1	1	0	1	0	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

^ : 字的或非运算。将两个字的对应位进行“或非”运算。

范例

Formula:

```
[t:0000]=41
[w:0020]=[w:0020]^([t:0000])
```

W0020

0	0	0	0	0	0	0	1	1	1	0	0	1	0	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

T0000

0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

运算后

W0020

0	0	0	0	0	0	0	1	1	1	1	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

~ : 字取非。将一个字的所有位进行“非”运算。

范例

Formula:

```
[w:0020]=~[w:0020]
```

W0020

0	0	0	0	0	0	0	1	1	1	0	0	1	0	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

W0020

1	1	1	1	1	1	1	0	0	0	1	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

2.6.4 Compare(比较命令)



and : “与” 运算

or : “或” 运算

not : “非” 运算

< : 小于

<= : 小于等于

<> : 不等于

> : 大于

>= : 大于等于

== : 等于

2.7 功能块

在写 D 脚本的时候，可以将一些常用的功能写成一个功能块，以便在其他 D 脚本中调用（GP270 不支持该功能）。

Call: 调用一个功能块。

New: 新建一个功能块。

Edit: 编辑选中的功能块。

Delete: 删除选中的功能块。

Copy: 复制选中的功能块。

Paste: 粘贴选中的功能块。

如何编辑功能块，请参照 D 脚本编辑器。



3 弹出键盘窗口

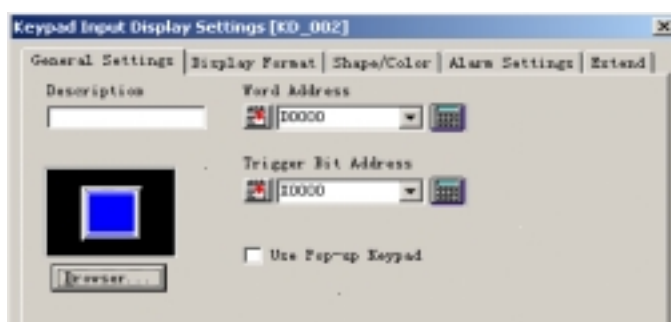
3.1 弹出键盘的应用

- ◆ 弹出键盘在实际应用中，可以使画面美观简洁紧凑。
- ◆ GP-PRO/PB V5.05 及以前支持不自动弹出键盘，可以用 U-TAG 实现弹出键盘。
- ◆ GP-PRO/PB V6.0 以后支持自动弹出键盘。

3.2 自动弹出键盘(v60 以后)

这是 GP-PRO/PB V6.0 以后版本的。

在[Parts（部品）]菜单中选择“[Keypad Input Display（键盘输入显示器）]”，出现“[Keypad Input Display（键盘输入显示器）]”设定对话框中（如图 2-2-1）。



(图 2-2-1)

在图上如果选择“Use Pop-up Keypad(用自动弹出键盘)”就可以实现自动弹出键盘功能。而在 GP-PRO/PB V6.0 以前的版本就没有这个选项。如不选择或没有这一选项，就要用 U-TAG 来实现。下面主要介绍如何用 U-TAG 来实现这一功能。

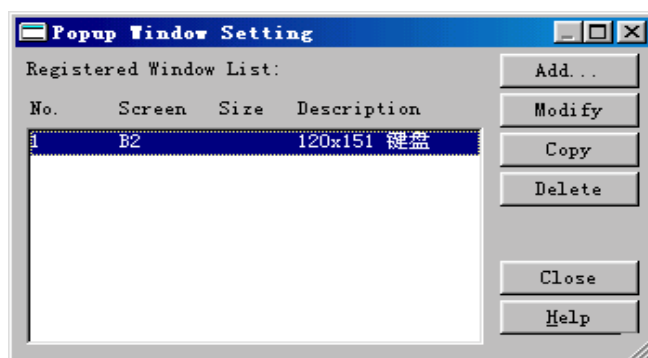
3.3 U-tag 实现弹出键盘的制作过程

1. 新建一工程文件，打开画面编辑器窗口。
2. 制作一键盘画面。新开一个基本画面，然后从[Parts（部品）]菜单中选择[Keypad（键盘）]，放置到画面，并保存画面（例如保存为画面 2），如图 2-3-1。



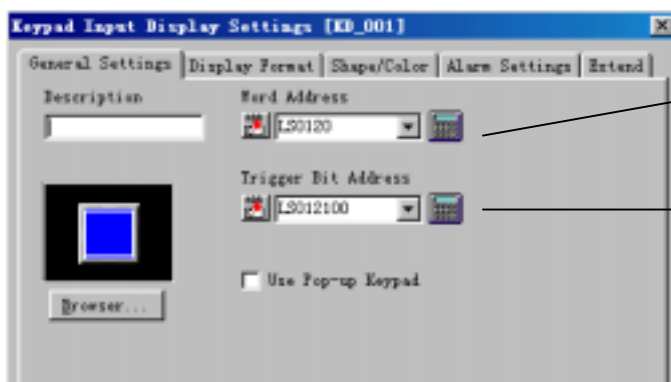
(图 2-3-1)

3. 将上面的键盘进行窗口注册，从[Screen（画面）]下拉菜单中选择[Window Registering（窗口注册）]，出现一个对话框（图 2-3-2），点击 ADD（添加）把键盘窗口注册为窗口，本例子注册为“窗口 1”（*具体的窗口注册请参阅《操作手册》第 98 页）。



(图 2-3-2)

4. 键盘输入显示。再新建一基本画面，在[Parts（部品）]下拉菜单中选择[Keypad Input Display（键盘输入显示器）]，出现如下对话框：



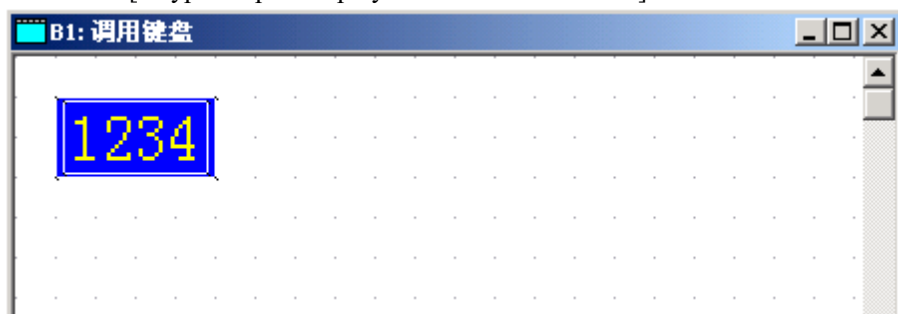
输入用于存储显示数
据的字地址

触发位地址

(图 2-3-3)

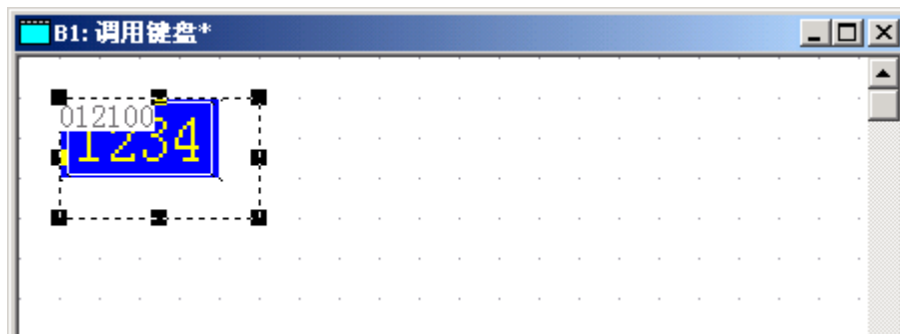
[Keypad Input Display（键盘输入显示器）]放置到画面后，画面如下：

图 2-3-4: [Keypad Input Display（键盘输入显示器）]放置到画面图



5. 按图 2-3-3 所示的设置地址例子，要输入数据时，须使键盘输入功能被激活，即使 LS012100 地址为 ON，可以在 “[Keypad Input Display（键盘输入显示器）]” 上面叠加一个 Bit Switch（位开关）（*要透明的），地址设置为：LS12100。如图 2-3-5

图 2-3-5：叠加一个透明的位开关



6. 用 U-Tag 去调用键盘窗口。从[Tags]下拉菜单选择[U-tag，弹出对话框]，然后进行设置。

U-Tag 有两种方式。

Direct 方式用一个寄存器(bit0)控制窗口的显示与消失，哪个窗口出现以及窗口显示的位置是固定的。Direct 方式下还可以选择 “High Speed” 方式，通过一个 Bit Address 来控制窗口的显示与消失。

Indirect 方式用四个连续寄存器控制窗口的显示与消失、窗口号、窗口 X 坐标、窗口 Y 坐标。因此可以改变窗口显示的内容、移动窗口的位置。

例子我们用 Direct 方式。

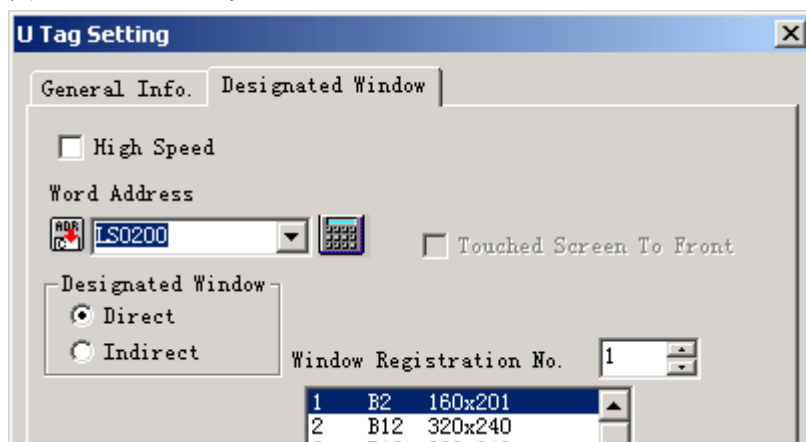
“Designated Window（指定窗口）” 里直接选择 Direct（直接）方式。

“Word Address（字地址）” 设置为：LS0200。

“Window Registration NO.(窗口注册号)” 选择 1，如图 2-3-6 的对话框。

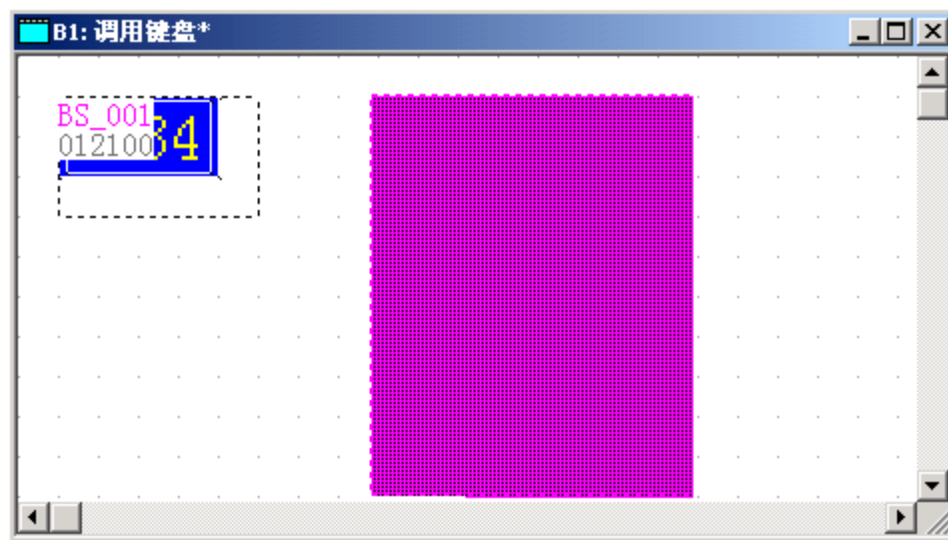
U-TAG 设置的对话框。

图 2-3-6：U-TAG 设置



设置好 U-TAG 并放置在“基本画面 1”上（自己认为比较理想的地方）。如图 2-3-7 红色区域所示。

图 2-3-7: U-TAG 的位置



7. 当然要想让 U-tag 动作，也要对其进行触发。

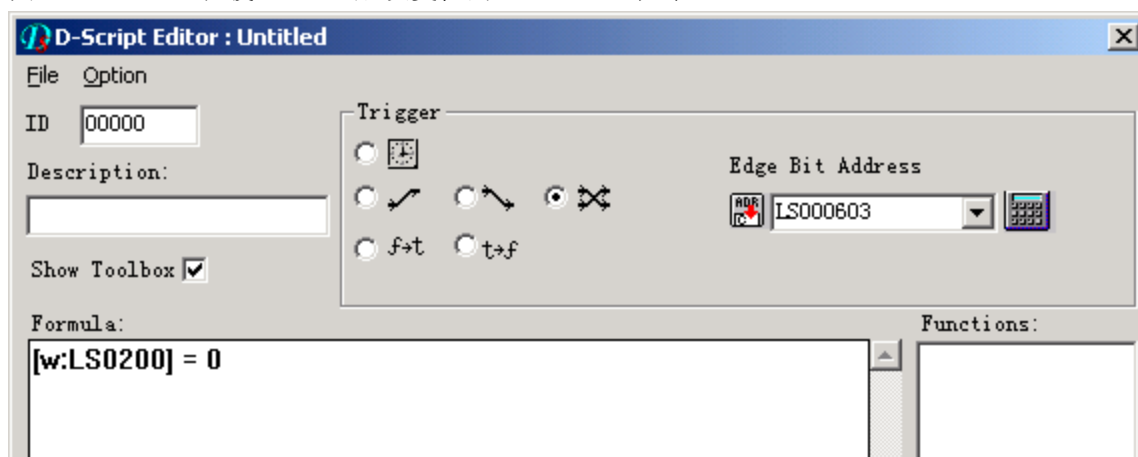
因为前面的例子，U-TAG 是 Direct 方式控制，所以在基本画面 1 中的“[Keypad Input Display（键盘输入显示器）]”上再叠加一个开关，按下该开关时将地址 LS0200 置为 1，控制窗口显示。

8. 弹出键盘输入数据以后，要让键盘窗口自动消失，只要把 U-tag 的触发位进行复位。

在 GP 系统数据区，Enter 键输入数据时，LS0006 的 bit03 位对应发生状态反转。我们可以根据这个 bit 的状态变化，增加一个 D-SCRIPT 程序，来使 LS0200 复位。注意选择 Trigger 为“上、下沿”。

此外，我们也可以在键盘 Enter 键的下面，叠加一个开关，按下该开关时将地址 LS0200 置为 0（当然这需要在键盘画面里做!）。

图 2-3-8: Enter 键使 U-TAG 触发复位的 D-SCRIPT 程序。



9. OK。

但是，还有可以简化的地方。如果 U-TAG 是用 High Speed 方式，即用一个 bit Address 来控制窗口的显示与消失。我们想到 “[Keypad Input Display（键盘输入显示器）]”也是用一个 bit Address 来触发输入状态的。因此我们可以公用同一个 bit Address 来控制输入、U-TAG。这样就省了 LS0200 地址的使用、以及对 LS0200 操作的相关设置！

但是 U-TAG 采用 High Speed 方式是有限制的。

3.4 窗口使用的限制

在使用 U-TAG 时当然也会有一些限制：

- 在同一窗口上不能同时执行 U-TAG K-TAG 及 V-TAG。
- 请勿在窗口上显示 Trend Graph（趋势图）。
- 有选择 High Speed 选项和没有选择 High Speed 选项的 U-TAG 不能在同一个画面显示。当选择 High Speed 选项虽然窗口尚未显示，但资料写入的 TAG（如 W-TAG, D-Script 等）触发发生时，将会动作。如果你希望先显示窗口后，再执行资料写入的 TAG，请不要选择 “High Speed”。
- 如果当前基本画面上的 TAG 和窗口数超过了 256，则第 257 开始的 TAG 不起作用（GP270/GP370/GPH70 为 128 个）。
- 再同一个画面中，最多可以设定两个使用 “High Speed” 的 U-TAG。
- 如果当前画面及它调用的窗口上，所有的 R-TAG 中设定的显示位置数超过了 512，则第 513 开始的位置不会显示（GP270/GP370/GP470 只有 256）。

4 密码制作

密码在实际应用中是很有用的，许多地方需要进行加密码保护，这样为了防止重要的操作参数被修改不便于机器工作。

下面主要进行介绍密码（及多级密码）制作过程：

4.1 一级密码制作的效果

举一个简单的例子：如下图 4-1 所示：

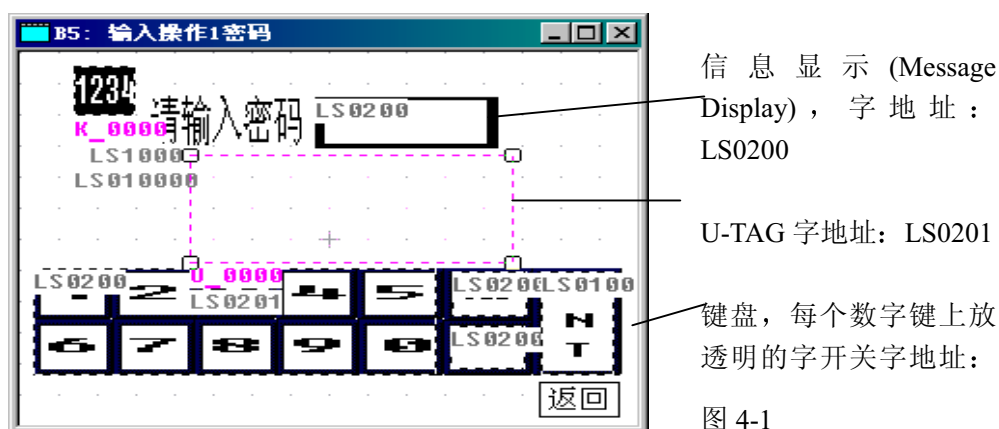


图 4-1

4.2 一级密码制作的具体介绍

1. 先建一基本画面，为了要输入密码，就要有一个键盘输入显示窗口。在窗口上放置一个 K-TAG(如图 4-2)就可以实现。本例中 K-TAG 的地址设置如下图 4-2，字地址设为：

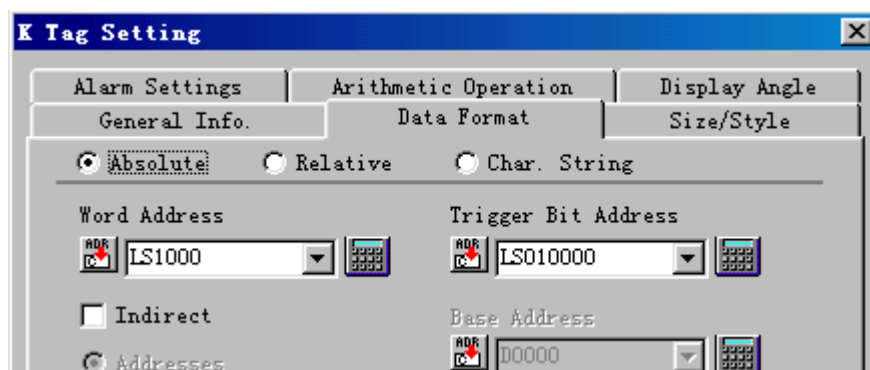


图 4-2

2. 当然要输入密码就要有个键盘，可以在部品中直接选择，形状可以自己按照需要改动。
3. 在实际输入密码时我们当然不想让别人知道，在输入密码时用*号来表示，这个功能可以用部品中的 “Message Display（信息显示）” 来实现。

本例为四位密码。“Message Display（信息显示）”的字地址设为：LS0200。用 LS0200 对键盘按键的次数进行统计。按一下键时 LS0200=1，就显示一个*号；按两下键时 LS0200=2，就显示两个**号。依此类推。

Message Display 的设定如下图 4-3 所示，图中 NO.of message(信息数)选择 8 以上都可以。message 所对应的每一条信息在 selected message 中输入，0 对应的信息输入空白内容，1/2/3/4 分别对应输入“*”；“**”、“***”、“****”。



图 4-3

4. 下面就要实现用 LS0200 对按键次数的计数。

- 1) 为配合*号显示，在键盘上叠加放置透明的开关按钮，对按键次数的计数。
- 2) 数字键上透明的字开关，按下时“次数加 1”，设置如下图（4-4），字地址是 LS200（同 Message display 相对应），按下时内容加 1（如果 1 个*号按一下就 2 个*号）；
- 3) DEL 键上透明字开关，按下时“次数减 1”，设置如图（4-5）；
- 4) CLR 键上透明字开关，按下时“次数清零”，设置如图（4-6）；
- 5) ENTER 键上的透明开关是为了实现输入密码后判断密码是否正确。ENTER 键上透明位开关设置，位地址：LS10001，



图 4-4

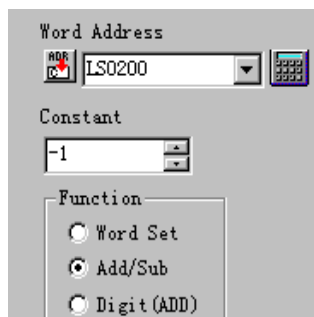


图 4-5

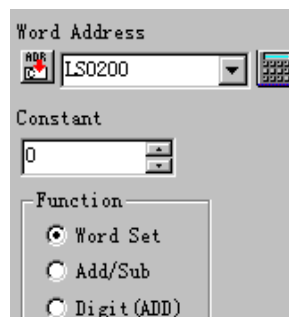


图 4-6

5. 当输入错误的密码时需要弹出一个报警对话框，提示密码错误。这可以用 U-TAG 实现（U-TAG 的使用在上一章中也有介绍）。

- 1) 先在另外的一个 BASE 画面里制作报警内容。
- 2) 将它注册为一个报警窗口如图 4-7。
- 3) 在密码输入画面，使用 U-TAG。U-TAG 的字地址：LS0201；放在画面 5 上自己认为可以的地方。



图 4-7

6. 相关的 D-SCRIPT 脚本

D 脚本编程是此例中最主要的，可以在基本画面 5 上，菜单栏选择 Special 的下拉菜单 D-Script，再选择 ADD，就可以进行编程了。

本例包括进入画面时初始化、密码判断、关闭报警窗口几个脚本。

◆ 进入画面时初始化。如图 4-8



图 4-8

图中当画面跳到基本画面 5（LS0000 是系统区地址它的内容显示的是画面号，本例是第 5 画面）时，信息显示（LS0200）为零，报警信息（LS0201 U-TAG）不触发，键盘输入显示（LS1000）为空，set[LS010000]使键盘输入处于待输入状态。

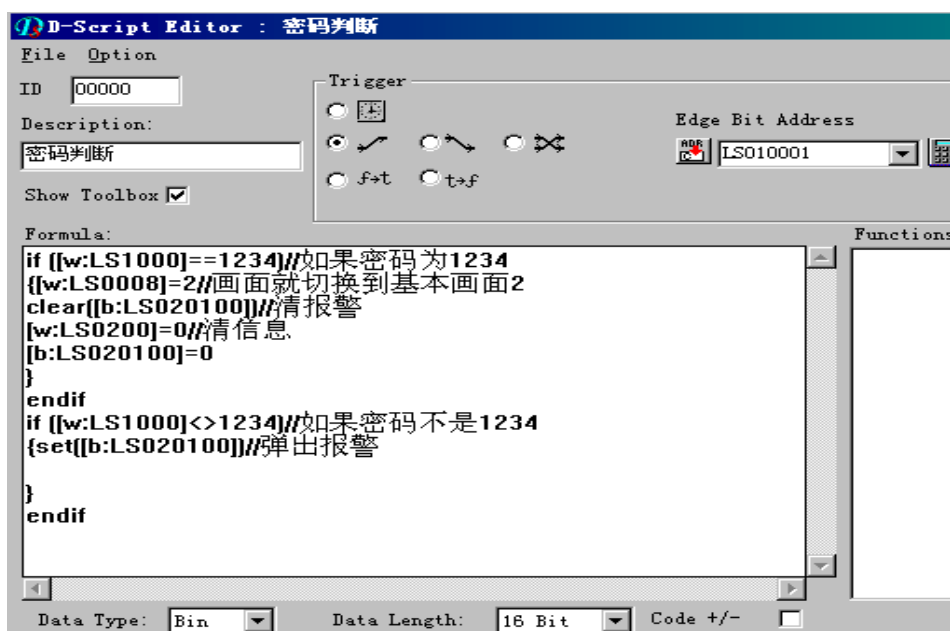
◆ 密码判断。如图 4-9

本例固定密码为：1234

```
if ([w: LS1000]==1234)//如果密码为 1234
{[w: LS0008]=2//画面就切换到基本画面 2
  clear([b: LS020100])//清报警
  [w: LS0200]=0//清信息
  [b: LS020100]=0
}
endif
if ([w: LS1000]<>1234)//如果密码不是 1234
{set([b: LS020100])//弹出报警
}
endif
```

*图 4-9 中 Edge Bit Address(边缘位地址)为：LS010001，同 ENTER 键上的透明的位开关相对应，当 ENTER 键按下后就执行这个脚本进行判断。

图 4-9



- ◆ 关闭报警窗口、清除信息、重新开始输入。如图 4-10



当密码错误、图 4-7 报警窗口弹出后，按下窗口里的 OK 键，消去窗口同时使信息显示 (LS0200) 清空为零。

至此，密码功能制作完成。

提示	1. 本例固定密码为：1234。如果要想在实用中采用可变密码，则将 1234 用数据寄存器号代替，寄存器的值就是密码。这样修改密码只需要修改寄存器的值。
----	--

4.3 多级密码制作

一个简单的多级密码就是在一级密码的基础上多加几条判断语句，下面就上例基础上举一个三级密码来进行说明。

1. 按上例一级密码放置部件，如图 4-1，地址及各设置都一样。
2. 要变化的主要是 D 脚本中的密码判断。其它的 D 脚本不变，把图 4-9 中 D 脚本的内容改为：

```
//一级密码判断
if ([w: LS1000]==1111)//如果密码为 1111
{ [w: LS0008]=2//画面就切换到基本画面 2
  clear([b: LS020100])//清报警
  [w: LS0200]=0//清信息
  [b: LS020100]=0
}
endif
//二级密码判断
if ([w: LS1000]==2222)//如果密码为 2222
{ [w: LS0008]=3//画面就切换到基本画面 3
  clear([b: LS020100])//清报警
  [w: LS0200]=0//清信息
  [b: LS020100]=0
}
endif
//三级密码判断
if ([w: LS1000]==3333)//如果密码为 3333
{ [w: LS0008]=4//画面就切换到基本画面 4
  clear([b: LS020100])//清报警
  [w: LS0200]=0//清信息
  [b: LS020100]=0
}
endif
//密码错误
if ([w: LS1000]<>1111and[w: LS1000]<>2222and[w: LS1000]<>3333)//如果密码不是
1111 和 2222 和 3333
{ set([b: LS020100])//弹出报警
}
endif
```

这样一个简单的三级密码就基本做好了。

5 多语言在线切换

5.1 多语言的应用

多语言显示切换功能需要在 GP377/77R/2000 系列上，GP-PRO/PBIII V6.0 版以上软件支持。

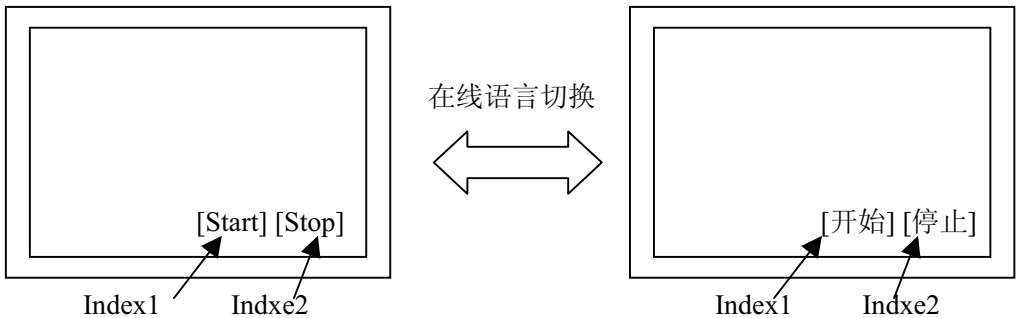
普通的文本内容或部件的标签，可以用字符串表的索引编号方式进行处理，这样可以非常方便地在运行时改变字符串表，从而实现多语言的在线切换，不用分别对不同语言重复做画面。

字符串表 1（英语）

1	Start
2	Stop
3	Running
	...

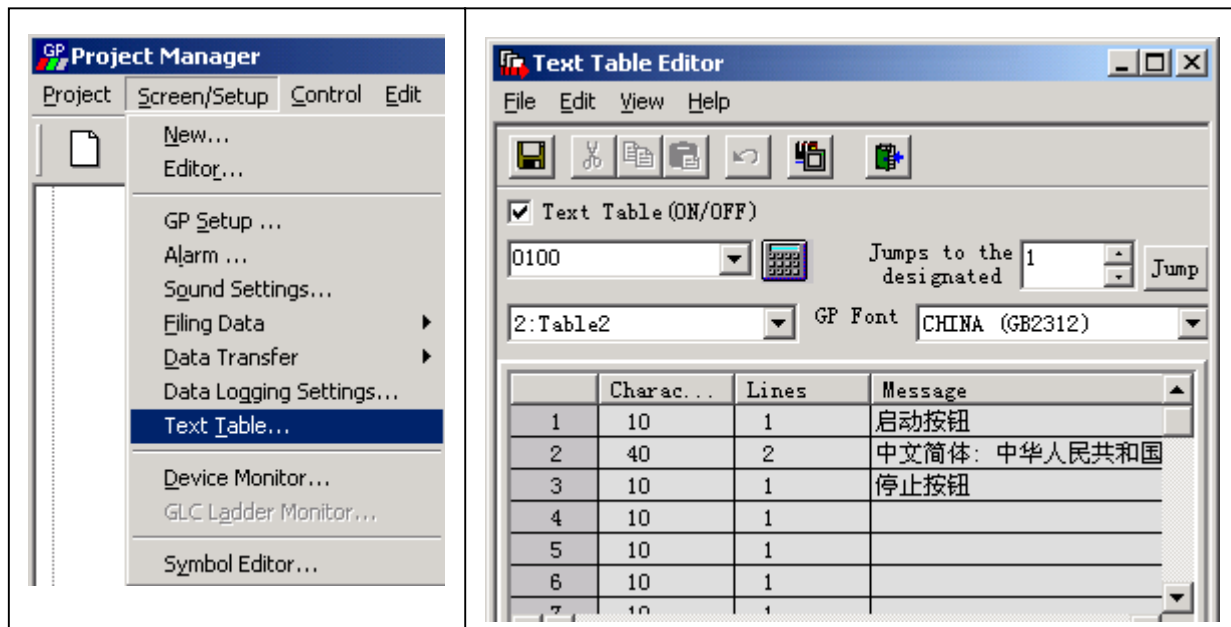
字符串表 2（中文）

1	开始
2	停止
3	运转
	...



5.2 多语言的字符串表

在工程管理器界面，选择[Screen/Setup] -> [Text table] 命令。出现“Text Table Editor”窗口。



首先，需要选中“Text Table (On/Off)”。

5.2.1 多语言表的控制地址

输入控制多语言表切换的控制地址，这是一个 Word Address，GP 运行时改变它的值就可以改变显示所使用的语言表。

例如上例中的地址：0100

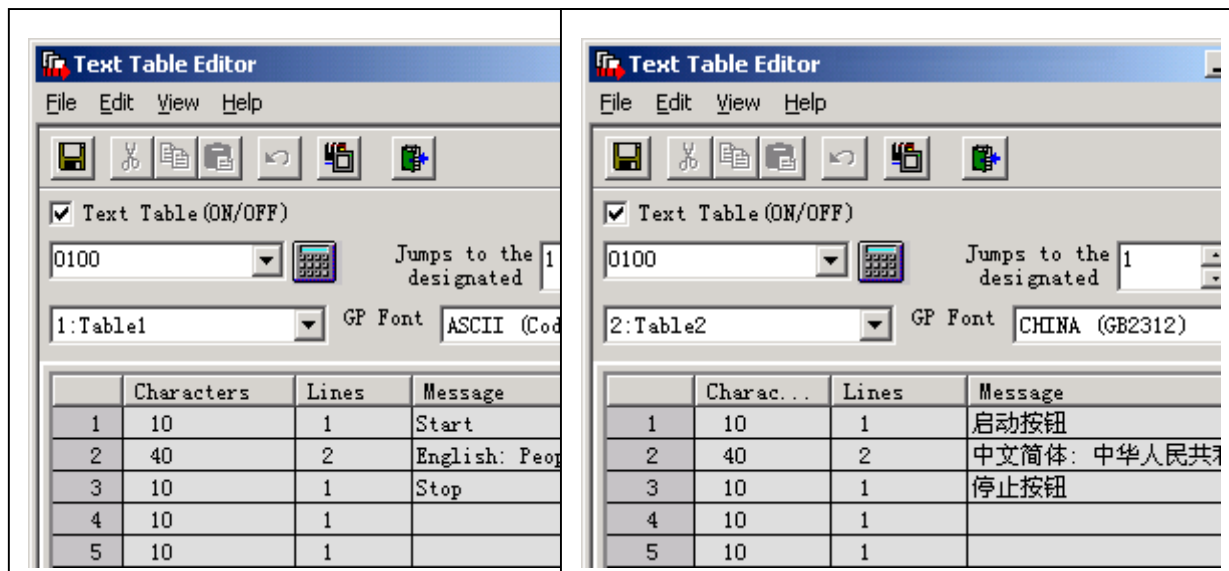
5.2.2 表的语言类型、名称

窗口左面有一个下拉选择框，分别对应 Table1—Table16。最多支持 16 种语言表。

■ 表的语言类型选择

下拉选择框选择每个 Table 表，可以在“GP Font”里选择相应使用的语言类型。例如：上图例中的，1:Table1 使用的是 ASCII（英文）；2:Table2 使用的是 CHINA（GB2312 中文）。

针对每种语言的 Table 表，需要逐条对应地在下面的表格中输入相应的文字。



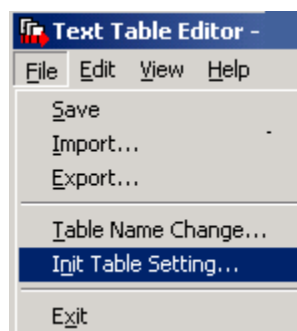
■ 表的名称

下拉选择框选择每个 Table 表，依次选择菜单[File] -> [Table Name Change]命令，可以改变表的默认名字“Table n”。



5.2.3 初始语言表的设置

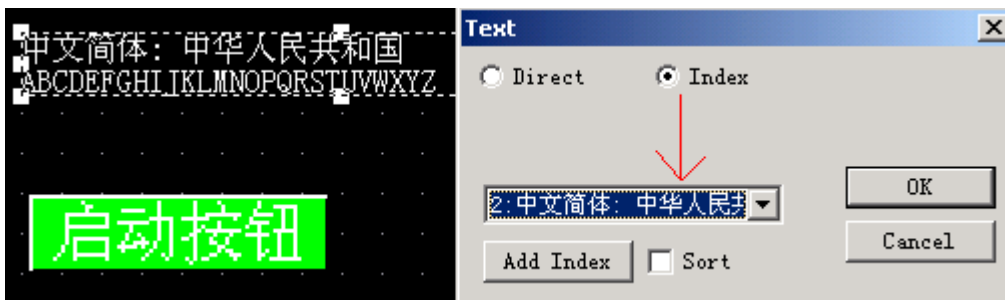
在“Text Table Editor”编辑窗口，选择[File] -> [Init Table Setting...]可以设置 GP 运行时的初始语言表。



5.3 编辑画面时使用语言表

在 BASE 画面里需要输入文字的地方，例如[Draw] -> [Text]，或者某个部件的“Label”下，就可以使用前面定义好的字符串表了。

- 例如下面输入的静态文本



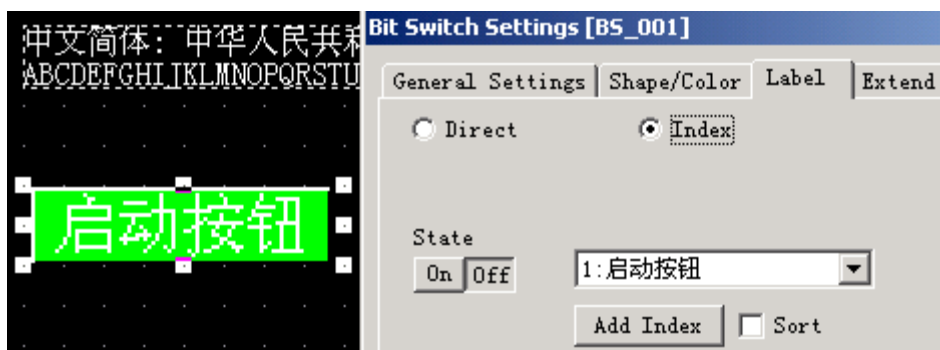
选择绘图静态文字时，出现“Text”属性对话框。如上图右边。

修改静态文字时，鼠标双击静态文字，出现“Text”属性对话框。

如果需要直接输入文字内容，可以选择“Direct”方式。这种输入的文字是不能进行在线语言切换的。

如果使用“Index”方式，就可以在下方的下拉列表框里选择前面定义过的字符串作为文字输入。因为它用这个字符串在语言表里的索引编号来对应的，所以运行时只要改变说使用的语言表，就可以进行多语言切换了。

- 例如下面输入的按钮的名称



选择绘图按钮标签时，出现“Bit Switch Settings”属性对话框。如上图右边。

修改按钮标签时，鼠标双击按钮，出现“Bit Switch Settings”属性对话框，切换到“Label”标签。

如果需要直接输入文字内容，可以选择“Direct”方式。这种输入的文字是不能进行在线语言切换的。

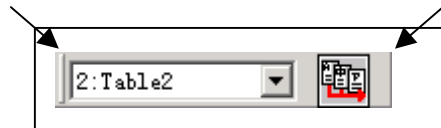
如果使用“Index”方式，就可以在下方的下拉列表框里选择前面定义过的字符串作为 ON/OFF 状态时的文字输入。因为它用这个字符串在语言表里的索引编号来对应的，所以运行时只要改变说使用的语言表，就可以进行多语言切换了。

- 编辑画面时的预览

依次选择[View] -> [Tool bar] -> [String Table]命令，可以在编辑窗口里打开/关闭以下的字符串表快捷工具条。

切换当前显示的语言表

启动“Text Table Editor”



通过这个工具条上的选择，可以快速地在画面上预览相应的语言画面。

5.4 运行画面时改变语言表

可以在“Text Table Editor”编辑窗口设置初始语言表。

前面介绍了设置多语言的控制地址，改变多语言表的控制地址的值，就可以进行语言表的切换。其值范围 1-16。例如前面的例子中，LS0100 的值为 1 时，以 Table1--ASCII（英文）显示；LS0100 的值为 2 时以 Table2--CHINA（GB2312 中文）显示。

具体改变控制值的方法很多。例如：可以做一个专门的语言选择画面，选择英文时，通过“WORD SWITCH”部件将常数“1”写入 LS0100；选择中文时，通过 WORD SWITCH 部件将常数“2”写入 LS0100。

5.5 多语言表的限制

以下内容是不能通过多语言表进行切换的。

- [A-tag]字符串显示功能
- [K-tag]字符串显示功能
- [S-tag]字符串显示功能
- [P-tag]格式化字符串显示功能
- [X-tag]显示功能
- 配方、数据记录等特殊部件。

6 CF 卡的使用

6.1 CF 卡的特性

0. 一个外部的存储设备是必需的，配方数据能存放在 CF 卡并且随时传送到 GP 中去。
1. 所有的报警数据，趋势图数据，采样数据和记录数据能够以*.CSV 文件形式保存。
2. 存储的数据能够通过数据库或其他的方法在 PC 上使用。
3. 画面数据能以*.MRM 文件保存，这种文件能够拷贝到其它的 GP 上。
4. 画面抓取适用于所有的 GP2000 系列。视屏画面抓取仅适用于 GP2500T/GP2600T 且需要配备 VM 单元。
5. GP77R 系列（需要一个适配器）和 GP2000 系列支持 CF 卡使用，GP2000 系列不另需适配器。

6.2 CF 卡使用注意

6.2.1 适配器是必须的

1. CF 卡只能用于 GP77R 和 GP2000 系列。使用 CF 卡时，需要配备适配器，GP2000 系列适配器就不需适配器了。
2. 被用于存储数据的 GP 单元必须提前设置 GP 系统和协议文件。

6.2.2 存放数据时需预先注意

1. 当 CF 卡被 GP 初始化时，文件档将自动生成。
2. 当配方数据中的“多个文件”设置没有使用时，文件夹的编号只有“1”。
3. 文件的编号能被设置成：从 1 到 8999（用配方数据时）和从 0 到 65535（用其它数据时）。
4. CF 卡数据文件名必须小于等于 8 个字节，这些文件名不能和 FAT32 文件一致。
5. 当存放新的数据到 CF 卡时，将覆盖旧的数据，CF 卡的可用空间必须大于存放数据的大小，因为在的数据被删除之前先写新的数据。
6. 在 GP 的系统设置（[GP System Settings]）中激活选中画面获取。

6.2.3 存取 CF 卡时需注意的事项

在 CF 卡和 GP 存取数据时发生下列动作将导致 CF 卡上的数据丢失：1）关闭电源；2）复位 GP；3）插入或拔出 CF 卡。为了防止数据丢失问题，建议预先在你的项目文件上准

备一个特殊的画面。这个特殊的画面只允许用于存取 CF 卡，防止 GP 误存取 CF 卡，这也将确保在你确认 GP/CF 卡的存取状态前不会任何发生任何动作。

这种存取监视画面如下面所述：这个画面使你能够检查数据是否正被装载到 CF 卡中去，确认这个画面没有数据正在被传送到 CF 卡中去后，你就可以关闭电源或复位和插拔 CF 卡。

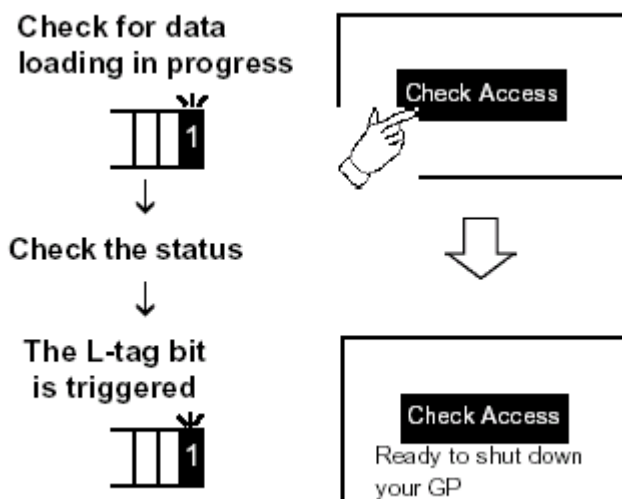
6.2.4 例子

GP 画面

- 1) 做一个主画面，上面做一个触摸键能使 PLC 中的某一位置 1 例如位 A，再放一个 L-Tag，它的触发位是 PLC 中的位 B，调用的画面是下一个步骤做的画面
- 2) 在另建一个主画面，在上面建一个库，写一条信息“Ready to turn the GP unit OFF”。

PLC 程序

- 1) 检查触摸键的位 A 是否已置 1。
- 2) 检查在[GP setup]-[Extended Setting]-[CF Card Data Storage Setting]设置的控制字，以确保没有数据在装载。当控制字的低八位为 0 时，数据装载没有被执行。
- 3) 确认没有数据在装载后，让位 B 置 1，调用 L-TAG，一条信息将出现在 GP 画面上。



6.2.5 其他注意事项

当 CF 卡中的数据被损坏时，数据不能被恢复。在这种情况下，你需要去重新格式化 CF 卡。记住，一定要尽可能的经常去备份 CF 卡上的数据，以防万一数据不能被读写。

用你的 PC 去读 CF 卡上的数据时，你的 PC 必须配备一个 PC 卡槽，把你的 CF 卡插入你的 PC 的 CF 卡的适配器中，然后再把这个适配器插入 PC 的 PC 卡槽中。

如果 CF 卡没有被插入 GP 适配器，就不要调用任何含有 CF 卡数据的工程项目。

GP-PRO/PBIII 数据的外部存储

使用 CF 卡（作为外部存储器）的步骤：

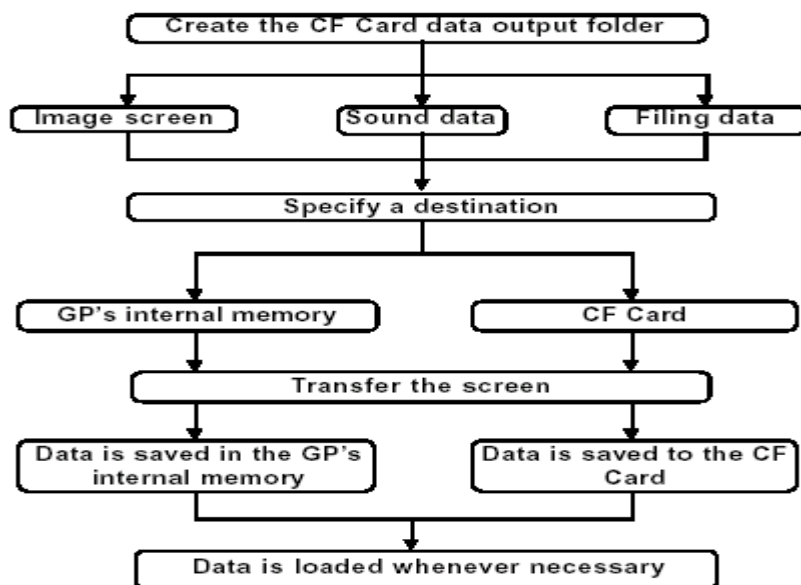
- 1) 准备建立一个 CF 卡数据输出文件夹（CF Card data output folder）。
- 2) 在做一个项目的时候，如果指定图像画面，声音数据和配方数据的数据存入 CF 卡，

这些数据都将被临时存入 CF 卡数据输出文件夹中。

3) 在传一个工程项目到 GP 中时, 存在 CF 卡的数据输出文件夹中的数据将被传入 CF 卡中, 其他的数据将被传到 GP 的内部存储器。

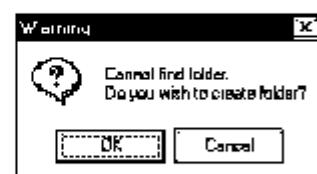
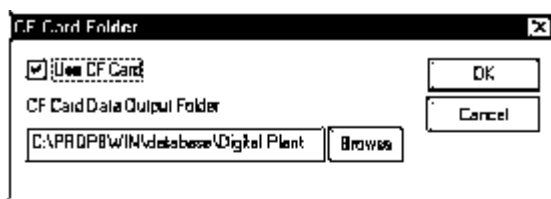
4) 在一般的 GP 操作中, CF 卡中的数据无论何时需要都可以被读取。

以下是流程图:



6.3 CF 卡数据输出文件夹设置

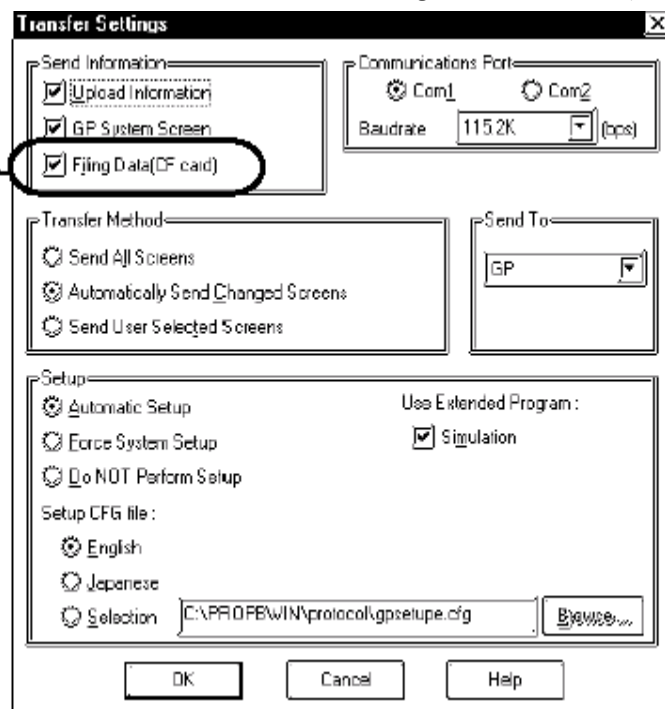
为了保存数据到 CF 卡, 你需要指明一个临时保存工程数据的文件夹的地方, 如果你指定 CF 卡作为存储图像画面数据, 声音数据或配方数据的目的文件, 那么它将被输出到指定的 CF 卡输出文件夹。从项目管理员的[Project]选择[CF Card Folder], 再选中[Use CF Card], 如下图所示, 指明一个存在的文件夹并输入它的名字, 或者点击 [Browse]按钮去选择文件夹。如果你在选择[CF Card Folder]之前已经指明 CF 卡作为目标, 下图所示的信息之一将出现。



6.4 输出文件夹传数据到 CF 卡

当工程文件被传到 GP 时，保存在 CF 卡输出文件夹中的数据同时被传送到 GP 的 CF 卡。为把你的配方数据传到 CF 卡，你必须在传送设置中选中[Filling Data(CF Card)], 如下图：

Transfers Filing Data to the CF Card



注意：

0. 当数据被传入 CF 卡，带有相同名称的文件将被覆盖，并将带有新名称的文件保存。如果工程数据改变并且传入 GP，以前传入 GP 的图象画面数据和声音数据文件仍然保存在 CF 卡。
1. 当“Memory Link Ethernet Type”被选做 PLC 的类型，画面数据不能传入 CF 卡。

6.5 选用 GP 的离线方式

这部分描述了通过 GP 的离线功能方式使用 CF 卡不同的功能：

0. 初始化 CF 卡
进入离线方式，选择“initialize”菜单后再选择“initialize CF”你就可以初始化 CF 卡。
1. 画面数据的备份
CF 卡的数据可以返回原来的 GP 和别的 GP，并且可以用这些数据。
2. 通过 GP 的离线方式选中“画面数据传送”的“COPY FROM CF CARD”，可以拷贝在 CF 卡上的文件。
3. 通过 GP 的离线方式选中“画面数据传送”的“COPY TO CF CARD”，可以在 CF 卡上生成一个备份文件。

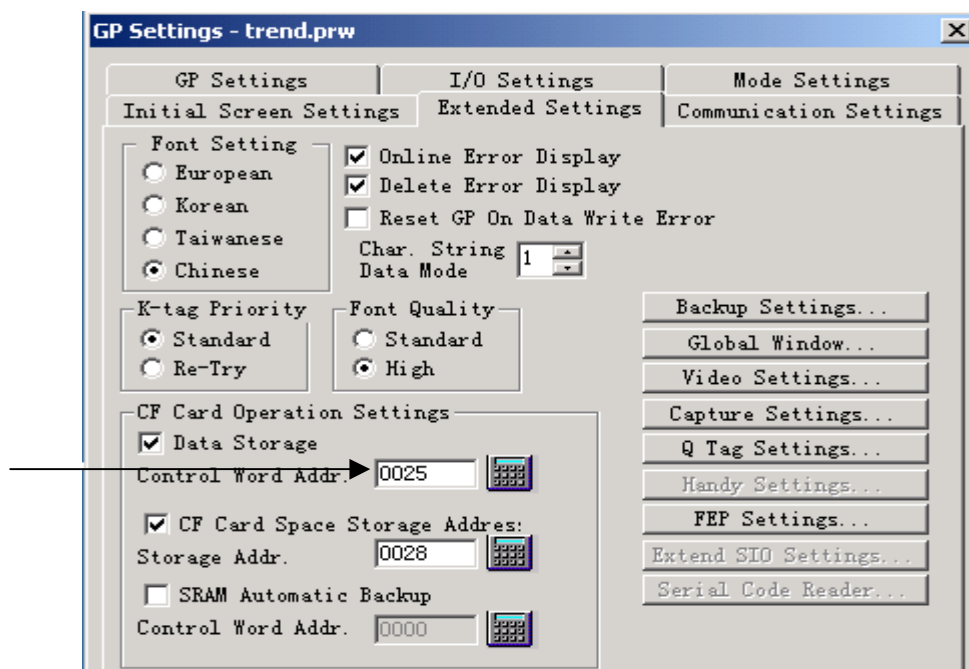
6.6 传送后备 SRAM 中的数据到 CF 卡

当 GP 单元工作时,任何存储在后备 SRAM 的报警趋势和登录数据和采样数据都能以 CSV 文件格式 (*.CSV) 存放在 CF 卡中,这些数据能通过赋一个数值到控制字而从 GP 的后备 SRAM 传到 CF 卡中去。通过选择 [Transfer]-[Upload Stored Data of CF Card] 或使用“CF Card Tool”,所有存放在 CF 卡中的 CSV 文件能装到你的 PC 中去并能作为数据使用。

为了把 GP 后备 SRAM 中的 CSV 文件传到 CF 卡中去,你需要把一个数值写到指定的控制字地址中去。

6.6.1. 定义一个字地址

控制字地址的选择详细说明见下图,那个箭头所指的就是控制字地址,



6.6.2. 传送数据到 CF 卡

控制地址被分成模式区和文件编号区,命令和状态码输入在模式区,文件名的说明在文件的编号区。

6.6.3. 模式区

当一个数被写到模式区地址后,相对应的操作将被执行,它的结果也被保存。

具体写的每个数相对应的功能描述见图：

Mode	Word Data	Description
Command	0001h	Filing Data
	0002h	Logging data
	0003h	Trend graph data
	0004h	Sampling data
	0005h	Alarm active/block-1 data
	0006h	Alarm history/block-2 data
	0007h	Alarm log/block-3 data
	0020h	Logging loop auto-save start
	0021h	Logging loop auto-save finish
Status	0000h	Completed successfully
	0100h	Write error
	0200h	Multi Unit not installed/CF Card not inserted/CF Card access switch OFF
	0300h	No data to be loaded (when no data is specified)
	0400h	File No. Error
	2000h	Logging loop auto-save responding correctly Control Address becomes this value during the auto-save mode. When the value is changed, the auto-save mode finishes.

6.6.4. 文件编号

- 1 （当使用配方，不选用多文件）
- 0~8999 （当使用配方，选用多文件）
- 0~65535 （当使用除配方以外的数据时）

注意：

当数据存放在 CF 卡中时，如果没有指定一个文件夹时，一个文件夹会自动生成并保存数据，如果此时没有文件夹自动生成（因 CF 卡没有初始化），一个写错误将发生。

如果一个写错误出现了，任何没有完成传输的文件将保留在 CF 卡中。

6.6.5. 记录数据循环自动保存

当使用循环特性时，CF 卡数据写功能将如下操作。当 GP 的文件满标志位 ON 的时候，记录数据将自动写到 CF 卡中去，文件格式是 CSV 格式。

1 开始自动存储

当用自动存储模式循环特性存储数据到 CF 卡中去时它指定文件编号并把“0020H”写到模式字中去。

当 GP 正常进入自动存储模式时，GP 把“2000H”写到模式字去。只有当值是“2000H”时，自动存储模式会继续，一旦这个值改变后，自动存储模式会停止。因此为了连续使用这种模式，不要改变这个值，要退出自动模式，请使用下面的退出命令。

2 退出自动存储模式

要退出自动模式，把“0021H”写到模式字中。当 GP 执行自动存储退出过程后。

GP 把“0000H”写到模式字中。当执行自动存储退出执行时，记录数据将被写到 CF 卡直到最后的后备 SRAM 数据。当开始或继续自动存储时，块名数据和记录数据从循环开始的地方写到 CF 卡中，而不是从预先存储状态的数据开始。

数据记录：（后备 SRAM）

（1）到（6）表示记录数据

在这个例子中，退出自动存储将导致记录数据写到（1）（2）（3）

(1)
(2)
(3)

在这个例子中，退出自动存储将导致记录数据写到（5）（6），当数据（1）到（4）被写了后，数据（5）（6）将增加进去。

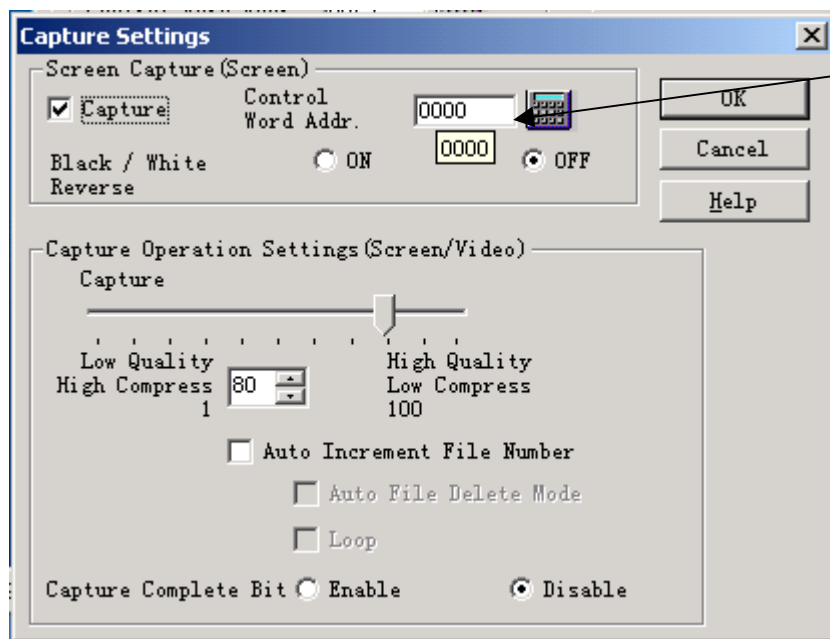
(5)
(6)
(3)
(4)

6.7 画面获取

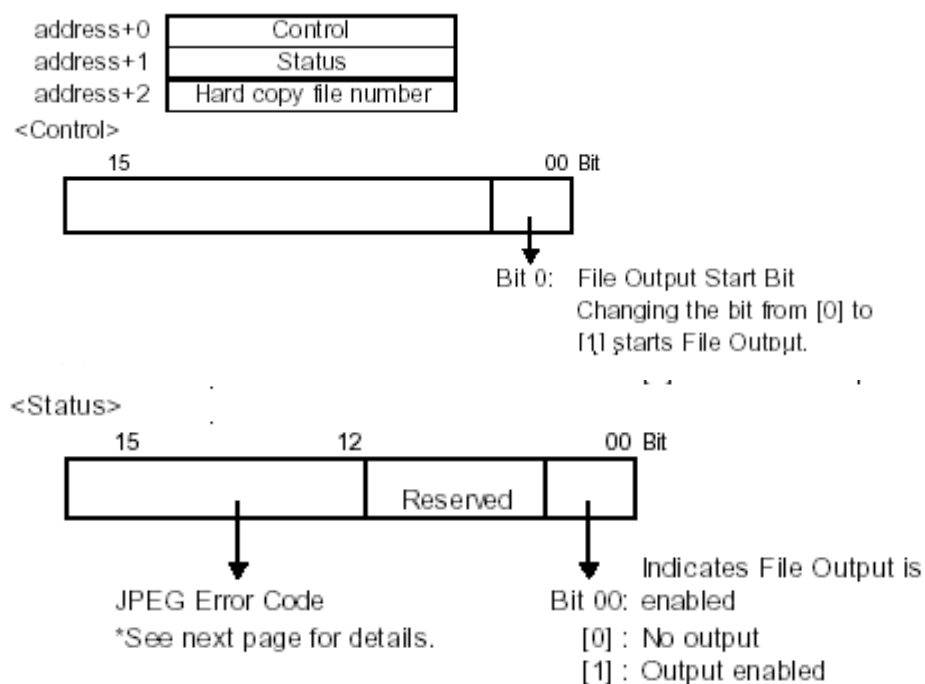
利用这个功能，在 GP 上显示的画面能以 JPEG 文件格式存放到 CF 卡（只有 GP2000 系列支持此功能）。

6.7.1. 控制字

为执行画面获取，用一个控制字地址来指明一个硬拷贝文件编号（0 到 65535），让控制字地址的“File Output Start Bit”置 1。控制字地址的设置 GP 的编辑软件“GP Setup”中选择“Extend Settings”，再选择“Capture Settings”就进入字的设置。看下面的图示，箭头所指的就是，



当“Auto Increment File Number”被调用，JPG 文件名以如下形式存储
 \CAPTURE\CPXXXXX.JPG (XXXXX 代表一个连续的序列号 0—65535)。
 可参考下图所示：



当使用“Auto Increment File Number”功能时，被存放的文件名是它的 JPG 文件数 (XXXXX)。
 <JPEG 错误代码>

Bit 12 to 15	Description	Details
0000	Normal Completion	Processing completed successfully.
0001	Reserved	
0010	Reserved	
0011	Reserved	
0100	No CF Card	Either no CF Card is inserted in the CF Card slot, or the CF Card cover is not closed completely during capture.
0101	CF Write Error	Either the CF Card has insufficient free space to hold the JPEG data, or no CF Card is inserted to hold the data being saved.
0110	Reserved	
0111	CF Card Error	CF Card not formatted.
1000	Reserved	
1001	File Number for Auto Increment is Exceeded	The file number exceeds 65535 and the Auto Increment function is enabled.

6.7.2. 画面获取的设置

获取质量:

设置范围: 1-100, 缺省值是 80, 1 是低质量最高压缩, 100 是最高质量最低压缩。

6.7.3. 自动增加文件编号功能

这个功能允许你自动分配一个连续的序列号给文件名。当一个画面被获取后, 它的文件名将自动的加一直至 JPEG 文件最大编号。GP 会在 GP 上电后, 开关 CF 卡和插拔 CF 卡后搜寻最大文件编号, 编号被写在控制字地址中的硬拷贝文件编号。

另外, 当 CF 卡容量不够或者文件数超过 65535 时, 使用 “Auto File Delete Mode” 允许画面被获取。











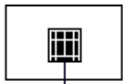
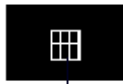



这个特性能自动的删除最小文件编号的文件并产生一个最高文件数加 1 的文件。例如, 文件 CP00100.JPG 到 CP00300.JPG 存放在 CF 卡上, 文件 CP00100.JPG 被删除, 而生成一个 CP00301.JPG 文件。但是, 当文件 CP00100.JPG 是只读文件时, 删除会被中断, 在状态区的 12—15 位被置 1, 并出现 “0101: CF Write Error” 信息。

当有一个 CP65536.JPG 文件存在时, 这个功能将删除所有存放在 CF 卡的 JPEG 文件且产生一个新的文件, 其文件名为 CP00000.JPG。

当 “Auto File Delete Mode” 禁用时, CF 卡没有足够的容量或有最大文件数时, 让 “File Output Start Bit” 置 1 也不能获取画面。

6.7.4. 黑白反相

设置这个特性是为了在 CF 卡上用黑白反相存放图象画面, 下面的图表描述了这种设置影响单色和彩色 GP 的结果。

GP-PRO/PBIII (PC screen)	GP Type	GP screen	Reverse Black & White Function (Within the CF Card)	
			Enabled	Disabled
 (White circle)	Monochrome	 (Normal) Black  (Reverse) White	 Black	 White
		 White	 Black	 White
 (Colors other than black and white) Example: Green	Color	 White	 Black	 White
		 Green	 Green	 Green

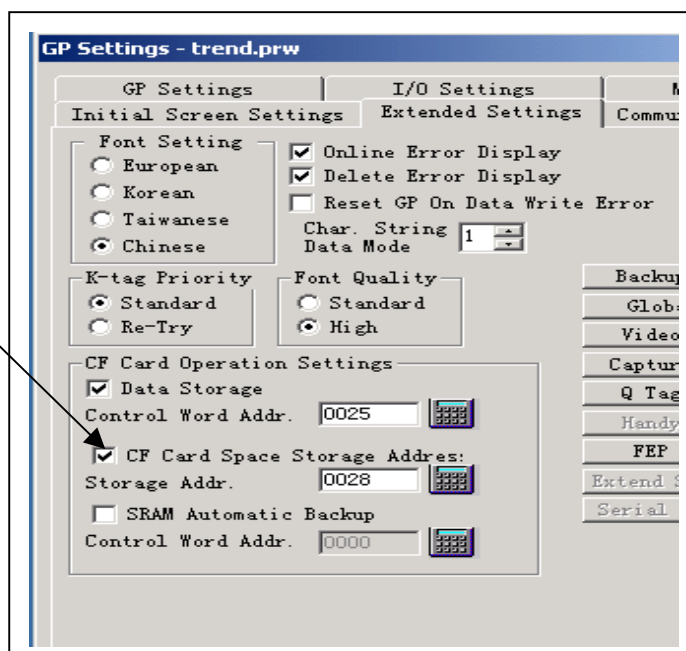
6.8 检查 CF 卡的可用容量

当前的 CF 卡的可用容量数目存放在 LS 区的指定地址内，通过检查这个地址中的值你就知道 CF 卡的可用容量，所有存放在 CF 卡的数据是以 KB 为单位的。在下面的三个例子中，CF 卡的可用容量不能正确检查且显示为零：

0. CF 卡的接受开关被关上了；
1. CF 卡没有插入；
2. 没有适配器。

注：放在 LS 区的 CF 卡的可用容量是近似值

CF 卡的可用容量存储地址设置如图，箭头所指方向就是

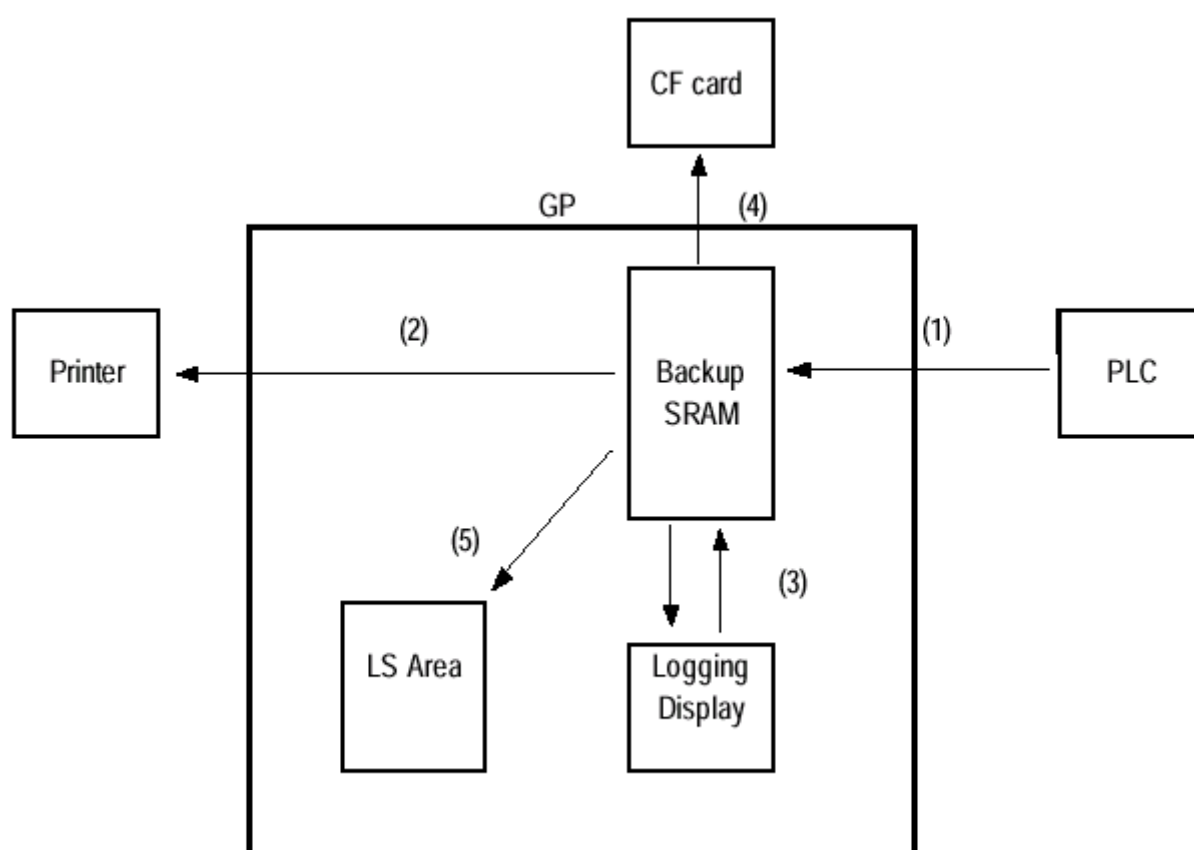


7 数据记录功能

7.1 数据记录的应用

数据记录功能可以定时或触发条件为 ON 时，采样 PLC 的数据，然后保存到 GP 的后备 SRAM 或 CF 卡中。记录的数据可以以表格形式显示、打印。显示时还具有求和、求平均、取最大、最小值功能。

数据记录可以完成下图的功能：



- (1) 按照触发条件进行数据记录。
- (2) 按照控制或者实时打印。
- (3) 在画面上显示记录的数据。
- (4) 根据控制，将数据保存到 CF 卡。
- (5) 根据控制，将数据传送到 LS 区。

7.2 数据记录占用 SRAM 的计算

■ 数据记录的结构、参数

记录	数据块 1	记录 1	寄存器 1	寄存器 2	寄存器 3	...	寄存器 32
		记录 2					
		...					
		记录 n					
	数据块 2	记录 1	寄存器 1	寄存器 2	寄存器 3	...	寄存器 32
		记录 2					
		...					
		记录 n					

通常方式下：

1. 每记录为 32 个 WORD 最大
2. $1 \leq \text{数据块数} * \text{记录数 } n \leq 2048$

■ 占用 SRAM 计算

占用 SRAM 的容量，不能超过各型号屏的 SRAM 容量，并且使用有优先顺序。

非 Loop 方式时：

占用的 SRAM (字节) = $20 + \{ (12 + 2 * \text{记录寄存器数}) * \text{记录次数} \} * \text{记录块数}$

Loop 方式时：

占用的 SRAM (字节) = $20 + \{ (12 + 2 * \text{记录寄存器数}) * \text{记录次数} \} * \text{记录块数} + (12 + 2 * \text{记录寄存器数})$

不同编辑软件版本可能稍有区别。实际占用 SRAM 的大小，可以依以下方法确定：

在工程管理器下，依次选择[Project] -> [Properties]，查看完整的工程信息，其中有关于 SRAM 的使用统计。

■ Economy 模式

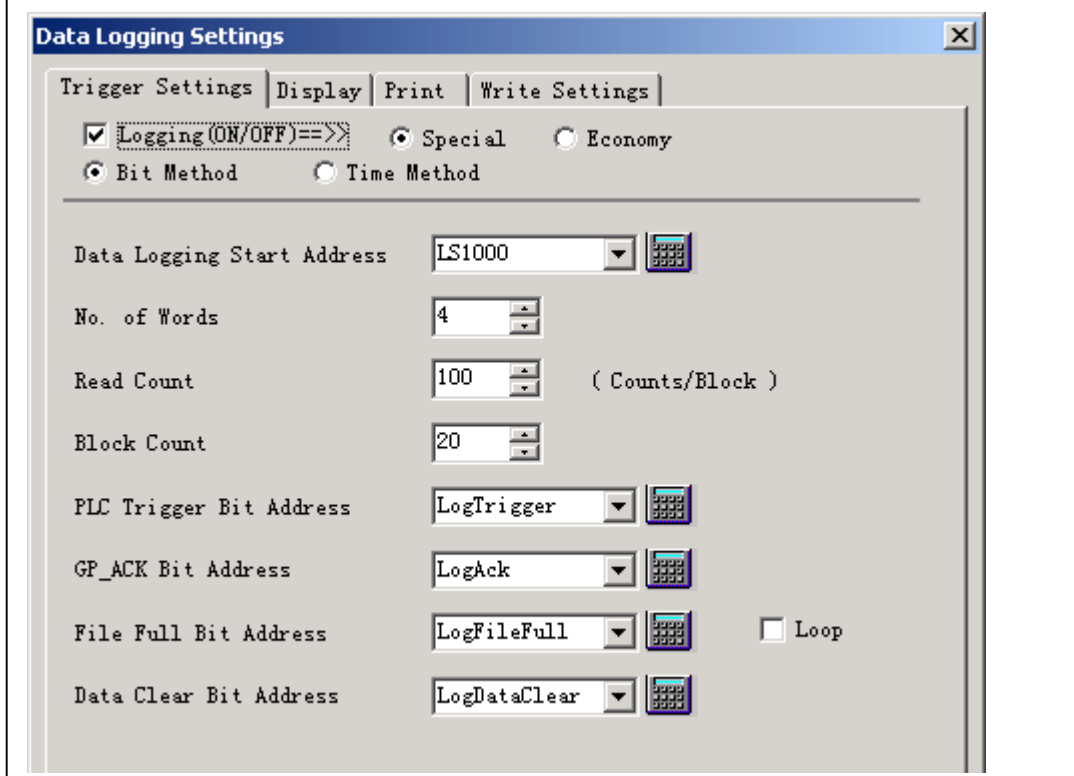
这种方式下，每记录最大可以指定 255 个 WORD 寄存器。这种方式下，只能使用 LOOP 方式。只有 GP2000 系列支持此功能。

7.3 数据记录的设定

7.3.1 触发方式的数据记录设定、记录时序

在工程管理器下，依次选择[Screen/Setup] -> [Data Logging Settings]，进行数据记录设定。

首先需要选中“Logging(On/Off)”，才能进行有关设置和应用，选择“Bit Method”方式。



在“Data Logging Start Address”指定被采样数据的起始地址。

在“No. Of Words”从起始地址开始、指定被采样的数据个数。最大为 32。

在“Block Count”和“Read Count”里分别指定最大采样记录的数据块数、每块的记录的次数。数据记录的内容可以分成多个 Block，满足： $1 \leq \text{Read Count} * \text{Block Count} \leq 2048$ 。

在“PLC Trigger Bit Address”中指定采样的触发地址。当 OFF->ON 变化时，PLC->GP 采样一次。

在“GP_ACK Bit Address”中指定每次采样的状态地址。当 PLC->GP 采样完成一次时，GP 将此设置为 ON 状态。

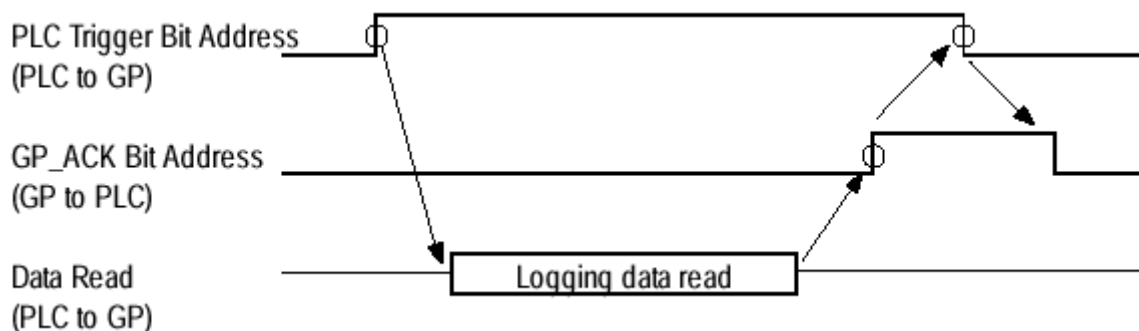
在“File Full Bit Address”中指定一个地址，用来指示采样记录是否达到了最大设定的次数，当达到了“Read Count * Block Count”的次数时，该地址被 GP 设置为 ON 状态。

在“Data Clear Bit Address”中指定一个控制地址，当 OFF->ON 时，所有后备 SRAM

中的记录数据都被清除。当所有数据都被清除后，GP 会将它设置为 OFF 状态。

“Loop”选项，用来控制当采样记录达到了最大设定的次数时，是否重新从第 1 条记录开始覆盖并继续记录。

■ 时序：



7.3.2 定时方式的数据记录设定、记录时序

在工程管理器下，依次选择[Screen/Setup] -> [Data Logging Settings]，进行数据记录设定。

首先需要选中“Logging(On/Off)”，才能进行有关设置和应用，选择“Time Method”方式。

The screenshot shows the "Data Logging Settings" dialog box with the "Trigger Settings" tab selected. The settings are as follows:

- Logging(ON/OFF):** Checked (ON/OFF) ==>>>
- Bit Method:** Radio button selected.
- Time Method:** Radio button selected.
- Data Logging Start Address:** D00000
- No. of Words:** 10
- Start:** 8 H 30 M
- Finish:** 15 H 55 M 0 S
- Duration:** 0 H 5 M 0 S
- Read Count:** 90
- Block Count:** 7
- Data Logging Auth. Bit Address:** X0000
- Block's Finish Bit Address:** X0000
- File Full Bit Address:** X0000
- Data Clear Bit Address:** X0000
- Loop:** Unchecked checkbox

在“Data Logging Start Address”指定被采样数据的起始地址。

在 “No. Of Words” 从起始地址开始、指定被采样的数据个数。最大为 32。

在 “Start Time” 里指定数据记录起始的时间。

在 “Duration” 里指定采样记录的间隔周期，可以是 15 秒的倍数。

在 “Read Count” 里指定采样记录的次数。根据 “Start Time”、“Duration”、“Read Count” 可以自动计算出 “Finish Time”。从 “Start Time” 到 “Finish Time” 即从开始时间到结束时间内记录的数据，成为 1 个 block。

在 “Block Count” 和 “Read Count” 里分别指定最大采样记录的数据块数、每块的记录的次数。数据记录的内容可以分成多个 Block，满足： $1 \leq \text{Read Count} * \text{Block Count} \leq 2048$ 。

在 “Data Logging Auth. Bit Address” 里指定一个采样控制地址。当该 bit 状态为 ON 并且到了采样的时刻，数据记录就被执行。

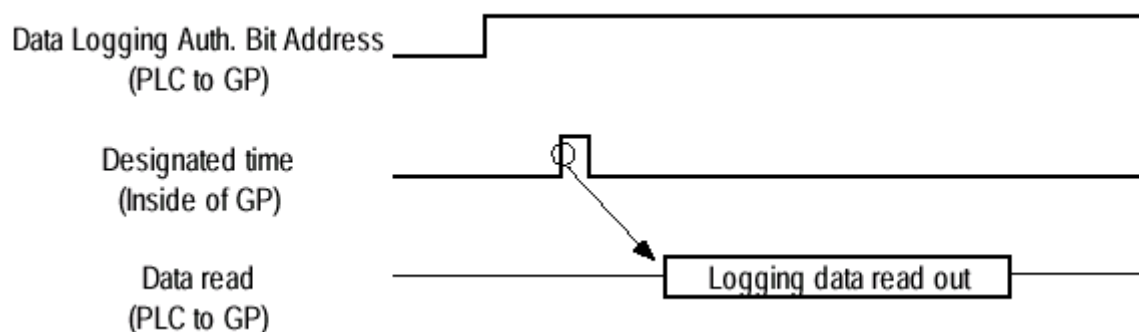
在 “Block’s Finish Bit Address” 里指定一个状态地址。数据记录每完成一次，这个 bit 被 GP 设置为 ON。当 PLC 检测到它的 ON 状态后，可以将它置为 OFF。

在 “File Full bit Address” 里指定一个状态地址。当数据记录达到了指定的块数，这个 bit 被 GP 设置为 ON。当 PLC 检测到它的 ON 状态后，可以将它置为 OFF。

在 “Data Clear Bit Address” 中指定一个控制地址，当 OFF→ON 时，所有后备 SRAM 中的记录数据都被清除。当所有数据都被清除后，GP 会将它设置为 OFF 状态。

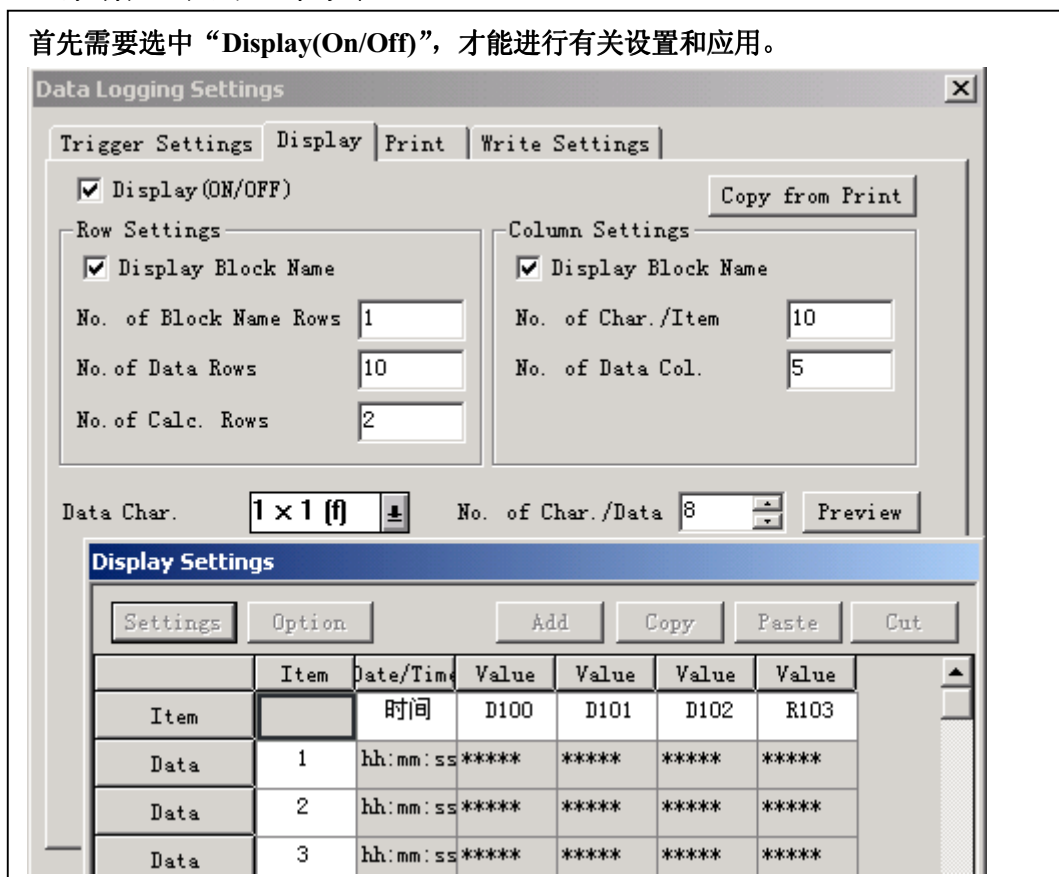
“Loop” 选项，用来控制当采样记录达到了最大设定的次数时，是否重新从第 1 条记录开始覆盖并继续记录。

■ 时序：



7.3.3 数据记录的显示设定

首先需要选中“Display(On/Off)”，才能进行有关设置和应用。



行设置 (Row Settings):

- Display Block Name: 是否显示采样数据的名称作为表头。
- No. Of Block Name Rows: 表头的行数。
- No. Of Data Rows: 数据显示的行数。
- No. Of Calc. Rows: 特殊计算数据的行数。

列设置 (Column Settings):

- Display Block Name: 是否显示名称作为表的第一列。
- No. Of Char./Item: 名称显示列的字符数 (宽度)
- No. Of Data Col: 数据显示列的列数

显示内容:

在“Display Settings”预览对话框里，根据以上设置会有相应行、列显示。

选择某行，可以设置该行是显示数据、或者文字。

选择某列，如果是日期/时间列，可以设置日期、时间显示的格式。

选择某列，如果是数据列，可以选择该列数据来源于数据记录的哪个地址，这个地址是以“数据记录起始地址 + 偏移量”来决定的。例如第一列，可以显示“数据记录起始地址 + 偏移量 0”，第二列，可以显示“数据记录起始地址 + 偏移量 1”。

7.3.4 数据记录的打印设定

首先需要选中“Print(On/Off)”，才能进行有关设置和应用。

Data Logging Settings

Trigger Settings | Display | **Print** | Write Settings

☒ Print (ON/OFF) ☒ Block Unit ☐ Realtime Copy from Display

Row Settings

No. of Data Rows:
 No. of Calc. Rows:

Column Settings

No. of Data Col.:

Header Footer Preview

Control Word Address:

Left Margin:

Print Completed Bit Address:

Print Settings

Settings Option Add Copy Paste Cut

	Char. Col.	Date/Time	Value	Value	Value
Char. Col.		时间	D100	D101	D102
Data	1	yy/mm/dd	*****	*****	*****
Data	2	yy/mm/dd	*****	*****	*****
Data	3	yy/mm/dd	*****	*****	*****
Data	4	yy/mm/dd	*****	*****	*****

Block Unit:

数据块打印方式。通过下面的控制地址来控制、并且有打印完成标记。在 Loop 方式下无效。

RealTime:

每项数据记录时，实时打印。

行设置 (Row Settings):

No. Of Data Rows: 数据的行数。

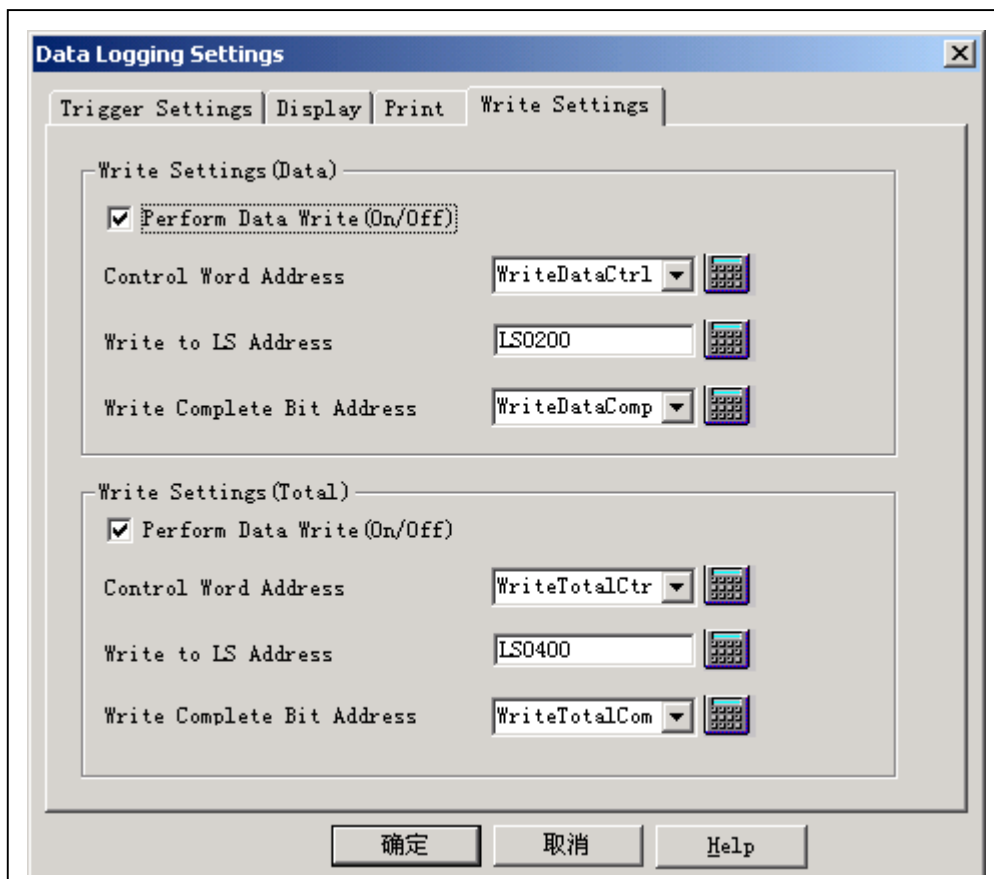
No. Of Calc. Rows: 特殊计算数据的行数。

列设置 (Column Settings):

No. Of Data Col: 数据列的列数

Header、Footer 可以分别编辑页眉、页脚。

7.3.5 数据记录的 LS 传送设定



- **Write Settings (Data):** 记录的数据写入到 LS 区。可以和 N-TAG 一起组合显示用。
Control Word Address: 数据写入 LS 区的控制地址。在需要数据写入 LS 区时，先指定一个块号 (Block No.)，然后改变控制地址的触发位为 ON。

	Bit 15	Bit 0
控制地址+0	保留 (全为 OFF)	触发位
控制地址+1	块 号	

Write to LS Address: 指定数据写入的起始地址。

Write Complete Bit Address: 指定位地址，当写入 LS 区完成后该位被置为 ON。

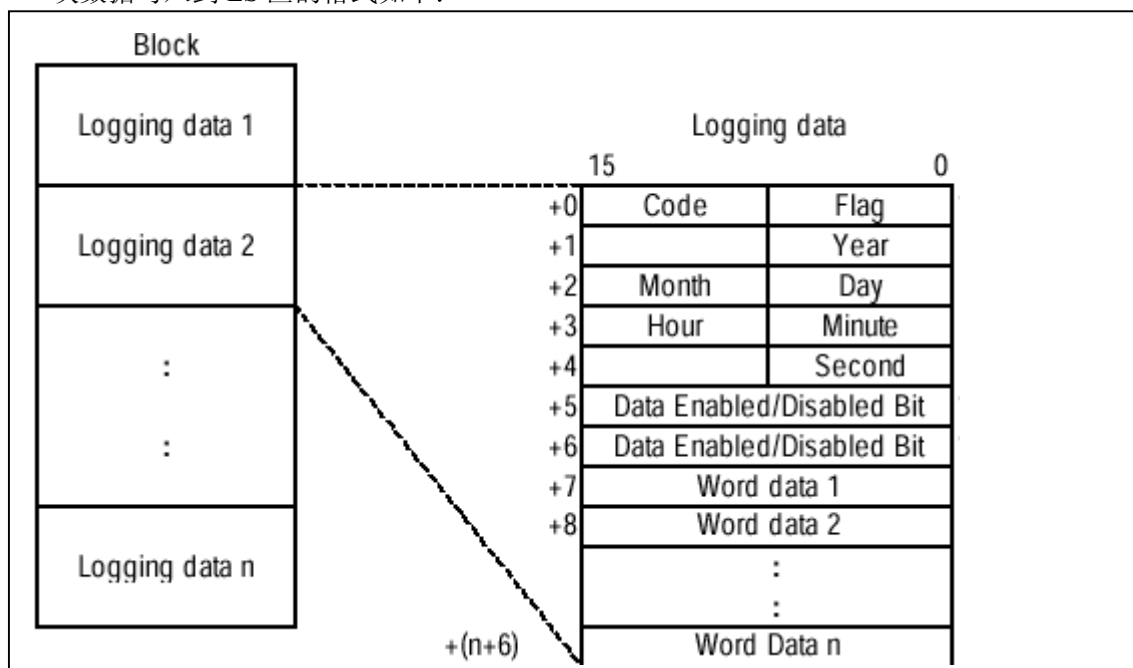
- **Write Settings(Total):** 将每块、每列的计算数据写入 LS 区。
Control Word Address: 计算数据写入 LS 区的控制地址。在需要计算数据写入 LS 区时，先指定一个块号 (Block No.)，然后改变控制地址的触发位为 ON。

	Bit 15	Bit 0
控制地址+0	保留 (全为 OFF)	触发位
控制地址+1	块 号	

Write to LS Address: 指定数据总和写入的起始地址。

Write Complete Bit Address: 指定位地址，当写入 LS 区完成后该位被置为 ON。

■ 块数据写入到 LS 区的格式如下：



其中：

标志的 bit0: 0 读未完成/1 读完成

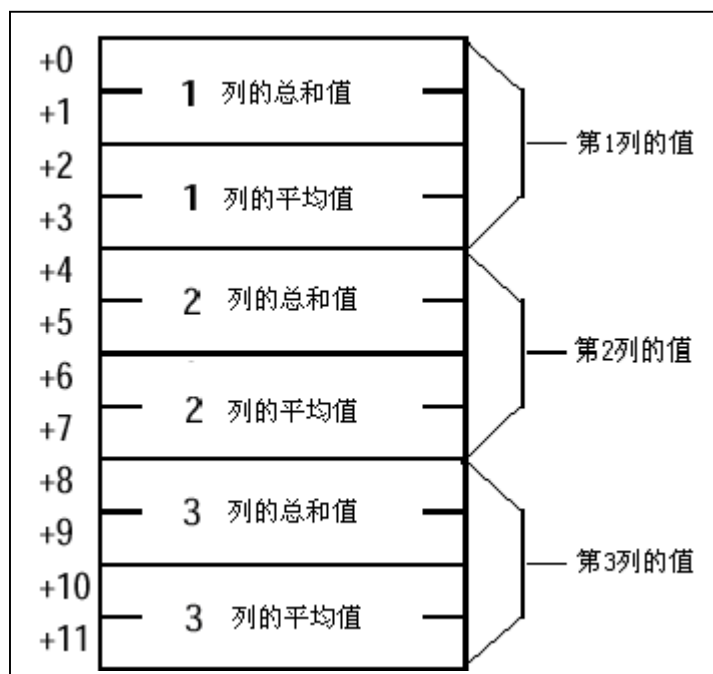
标志的 Bit8-15: 0 正常完成/1 读错误

日期、时间：都是 BCD 格式的最后 2 位数。

Data Enable/Disable: 每 1bit 对应 1 个数据是否有效，共 32bit（最大记录 32 字）。

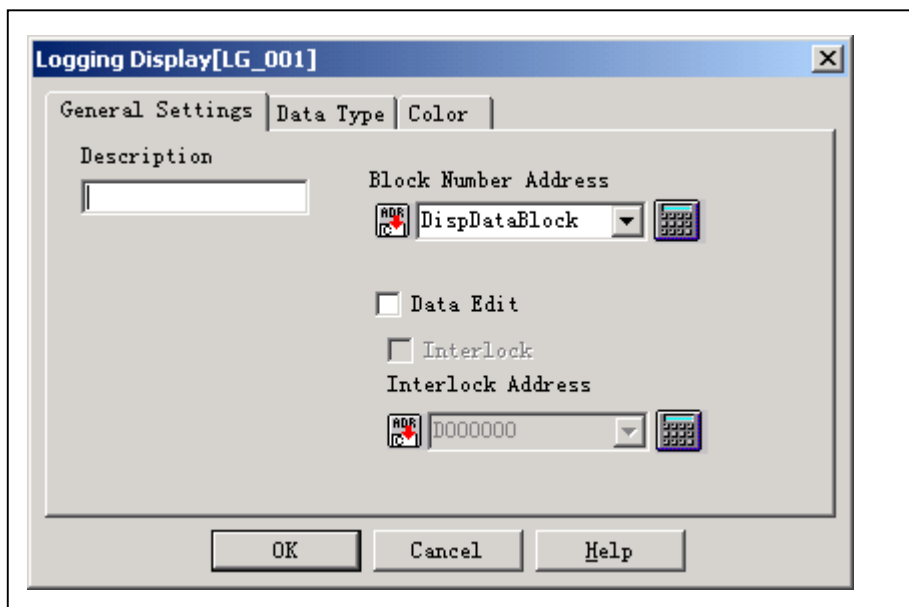
■ 计算数据写入到 LS 区的格式如下：

计算数据（总和、最大、最小）都是 32bit 数据。



7.3.6 数据记录在画面上的显示设定

在画面编辑器下，依次选择[Parts] -> [Logging Display]，进行设定。



在[General Settings]标签下：

Block Number Address：指定数据显示的块号。

Data Edit：允许数据编辑。

在[Data Type]标签下：

指定数据显示的行列数，以及单元格。

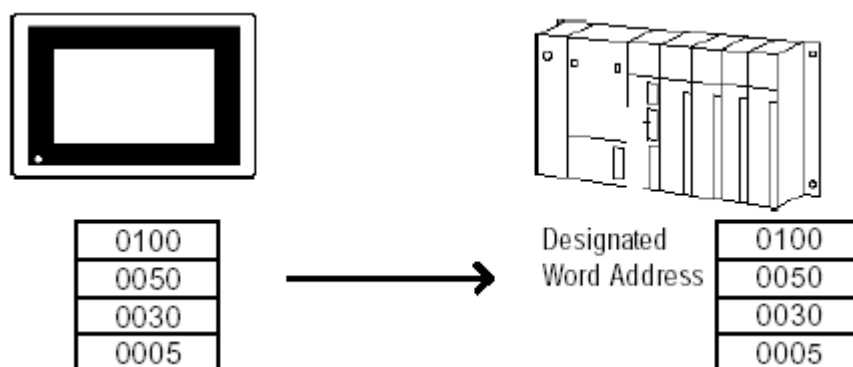
在[Color]标签下：

指定显示字符的颜色。

8 配方功能

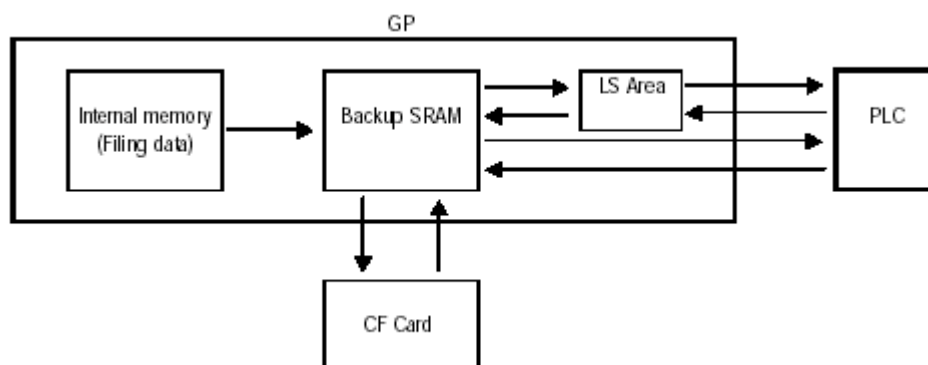
8.1 概述

配方功能是用于传输初始化数据（条件参数、框架数据等）给 PLC。配方可以被用于对机械操作进行设置等。配方功能的数据被称作文档数据。



8.2 配方功能

1. 设置文档数据时，选择工程管理器中的[Filing Data]/[Filing List]选项。
2. 两种方法把文档数据传输到 PLC 中：第一是自动传输，通过 PLC 的触发把文档数据传输到 PLC 中；另一种是手动传输，GP 屏幕上选定数据再传到 PLC 中。
3. 用手动数据传输时，从工具箱或部件菜单中选择文档数据显示。
4. 文档数据能用每个工程文件号来控制。
5. 文档数据可以被保存在 GP 的内部存储器中或 CF 卡中。
6. 文档数据存储在 PLC 的连续地址中。
7. 文档数据能够从 PLC 中读出并传输到 GP 的内部 SRAM 中。一旦文档数据被读出，它又可以被写入 CF 卡中。
8. 设置文档数据因要通过 LS 区传输数据并允许数据在 GP 屏上编辑及写入 PLC 中。
9. **文档数据传输流程：**文档数据首先从 GP 内部存储器或 CF 卡被写入 GP 的后备 SRAM 中，然后通过直接方式或经由 LS 区被传输到 PLC 中。

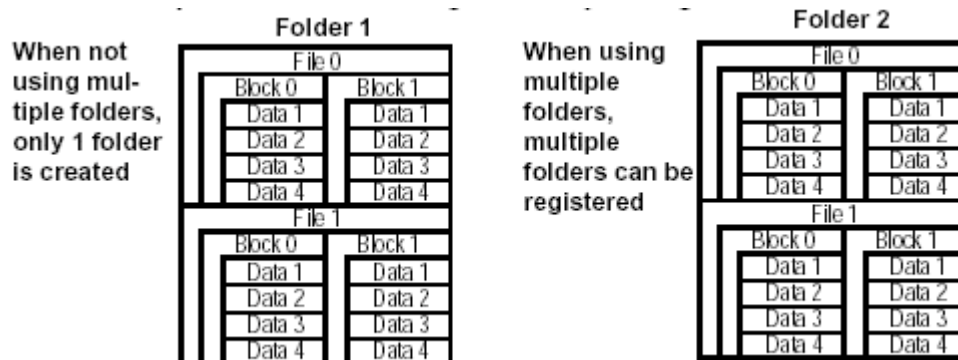


文档数据传输流程图

10. 文档数据结构

文档数据有文件夹、文件、模块和数据四种形式，还被组织成其他可能的形式。

多种文件夹能用于 GP 的内部存储器或 CF 卡中来寄存文档数据项目。



1)、不是多种文件夹使用时：只有一种文件夹被寄存时，这种情况下，文件号通常不被指定，当必须有时，可以设定它的值为“1”。

2)、多种文件夹使用时：多种文档数据夹被寄存。在 GP 中，文件夹号需要被寄存来传输数据到 GP 的内部存储器和后备 SRAM 中。即使一个文档数据夹能被 GP 用上一次，也就要使用这个文件夹。

- 多种文件夹可以被登记到 GP 的内部存储器或 CF 卡中。
- 文件夹数不大于 64 时，数据登记在 GP 的内部存储器中，当文件夹数大于 64 而小于 8999 时，数据登记在 CF 卡中。
- 选择一个文件夹在 GP 上使用。
- 只有一个数据文件夹能够传输到 GP 的后备 SRAM 中。
- 当多种文件夹登记时，文件被标号。文档数据结构与现有的数据结构相同。另外，如果先前创建的文件夹存在于 CF 卡中，则这个文件夹将被自动命名为“1”。

11. 已注册文档数据

当保存文档数据时，不超过 2048 个文档数据文件能被登记，并且在每个文件中能有不超过 1650 个数据模块，及在每个数据模块中能有不超过 10000 条数据（然而当数据格式为 32 位时，只能有 5000 条数据被登记）。

12. SRAM 存储空间计算

1 个文件容量=96（定值）+（32+2（4）*数据条数）*数据模块数（对于 16 位字节设备用“2”，而 32 位字节用“4”）。

所有文件容量为 59526。

以下所示为 16 位字节数据格式下存储空间对应表

No. of files	No. of data pieces	No. of data blocks	Total capacity of all files
1	2	1650	59496
1	640	45	59136
1	10000	2	40160

13. 后备 SRAM 使用优先权（后备 SRAM 的使用从第一条开始）

- 1)、报警履历显示
- 2)、数据采样
- 3)、趋势图表
- 4)、后备 LS 区
- 5)、数据记录
- 6)、文档数据

在下列情况下后备 SRAM 中的数据将被丢失：

- 1)、GP 存储器初始化。
- 2)、数据传输
- 3)、GP 系统和协议设置。
- 4)、GP 自诊断被执行。

5)、GP 的离线方式被设定，然而，如果 GP 的系统译本为 1.3 版本或更高版本，后备 SRAM 数据将不被丢失。

6)、这些数据的注册限度随着 GP 的不同而不同，例如：后备 SRAM 的使用与当前设置。

7)、不超过 59520 字节的文档数据能够被存储在 GP 的内部存储器的一个文件夹中。后备 SRAM 中所存文档数据的最大容量大约为 95K 字节，它能够存储在 CF 卡中（只有文档数据存储在后备 SRAM 中时）。

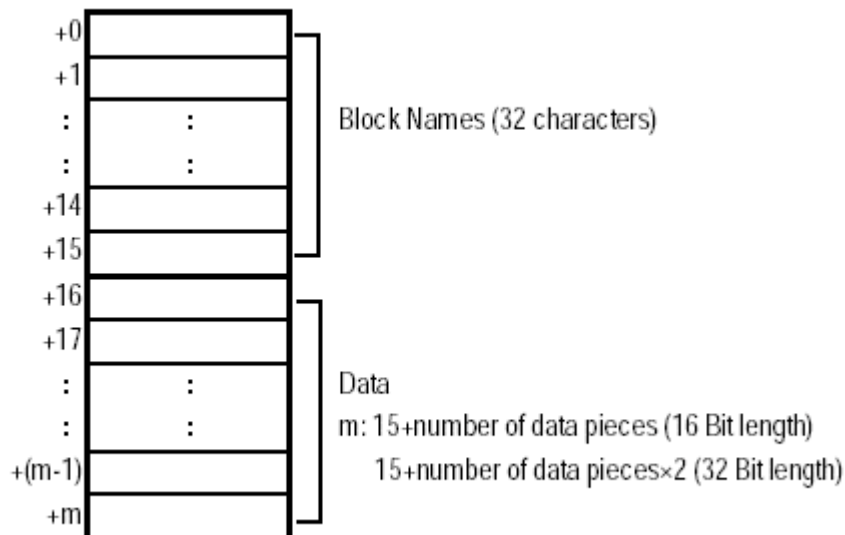
8)、当大量的数据地址建立时，写数据到 PLC 中需要更多的时间，可能要许多秒或许多分钟。

9)、当数据正在写入时，屏幕上有显示，例如：tags，它可能不会更新或消失得很慢，当屏幕有变化，且为 Q-tag（报警履历显示）时，它也可能执行得不慢。如果在这时主界面或窗口界面有变化，那么屏幕上的数据，例如 tags 数据将被读出，而且写数据到 PLC 中去的执行速度将比正常时的慢。

不要在同一时间内执行以下的操作：

- 1)、数据在 LS 区和 PLC 之间传输。
- 2)、数据在 SRAM 和 PLC 之间传输：通过文件名显示。
- 3)、数据在 SRAM 和 PLC 之间传输：通过传输控制字地址的设定。

Designated address



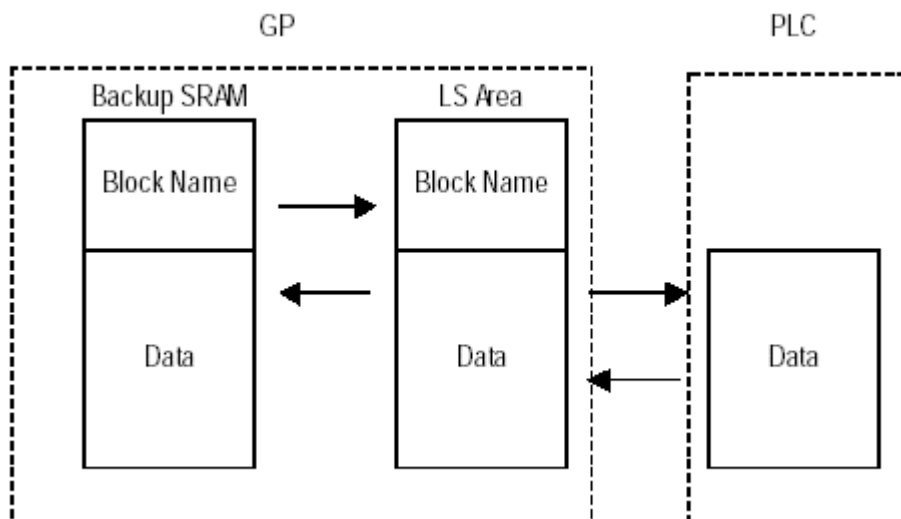
在当前数据传输完成后，确定对执行数据的传输。

在当前数据写入时，随后数据的写操作将不能被接收。

14、LS 区文档数据结构

15. 数据传输过程

数据在后备 SRAM 和 LS 区之间传输时将写在已有的模块名和块数据之前，数据从 PLC 传输到 LS 区时将写在文件数据和先前模块名之前。



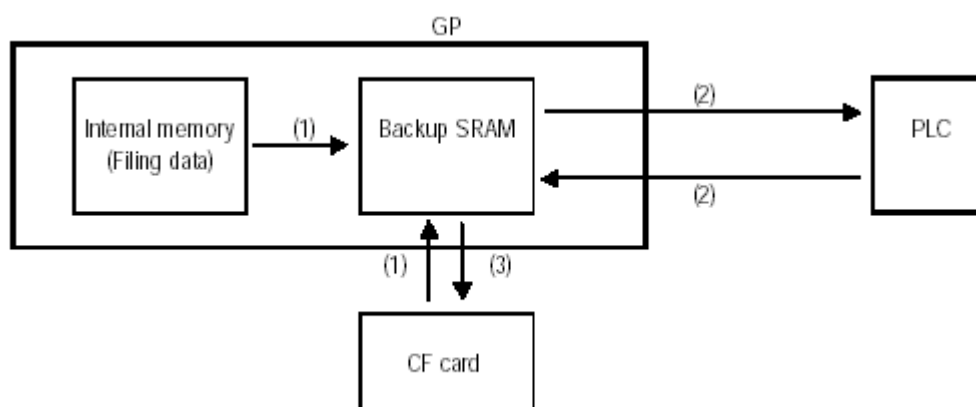
当设备用的是 32 位字节系统时，文档数据不能经由 LS 区传输到 PLC 中。

16. 文档数据传输方法

文档数据传输方法大致可以分为以下三种类型：

<1>. 自动传输

经 PLC 的触发，文档数据从 GP 传输到 PLC 中，用控制字地址来代替利用文档数据显示在 GP 与 PLC 之间进行数据传输。



当使用单个文件夹时，指定文件夹号为“1”。

如果指定文件夹号不存在，数据将不能传输到后备 SRAM 中，另外，写完成后，位

地址将不被置成“ON”状态。

当数据不被传输到后备 SRAM，GP 内部特殊继电器（LS2032）的第 9 位将置成“ON”状态。

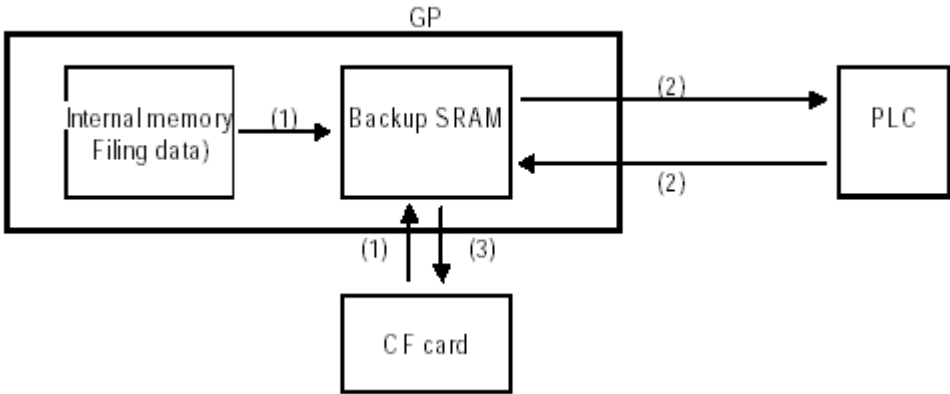
文档数据在 CF 卡中的存储命名如下表所示：

Folder Setting	File Name
Use Multiple Folders: OFF	ZF00001.BIN (fixed)
Use Multiple Folders: ON	ZF****.BIN (**** = Folder No. ex: ZF00001. BIN, ZF00002. BIN,etc.)

＜2>. 手动传输—类型 1

文档数据通过触摸 GP 屏幕上的触摸键传输到 PLC 中，利用文档数据显示数据传输到/出 PLC，通过“PLC-SRAM”转换键或“SRAM-PLC”转换键实现。

当文档数据从后备 SRAM 发送回 PLC 时，新的发送数据将覆盖已有的数据。为了

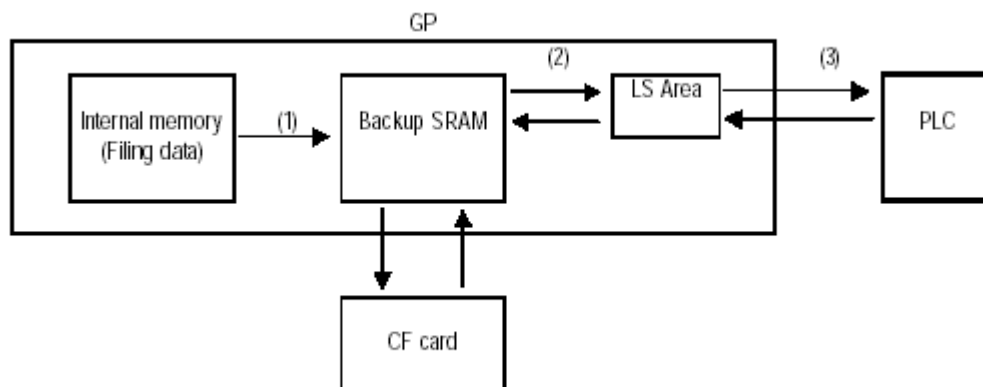


防止这一现象，预先要创造存放有用数据的另一个数据模块从而能继续发送数据，这个模块通过控制字地址来指定。

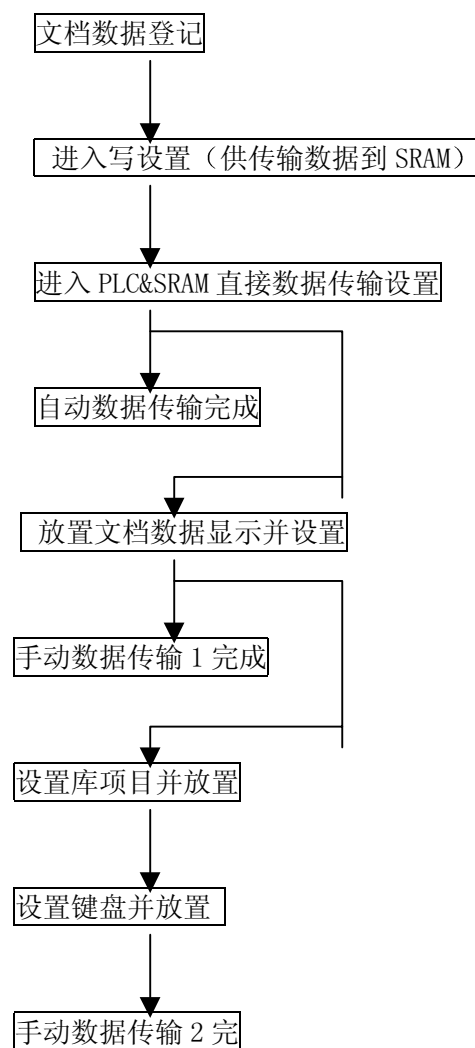
File 0			
A	B	A Return Value	B Return Value
50	40	0	0
60	30	0	0

＜3>. 手动传输—类型 2

经由触摸屏上的触摸键来选择需要的文档数据，执行数据的变化，且传输数据到 PLC 中，利用文档数据显示传输数据到/出 PLC，通过“LS-PLC”触摸键或“PLC-LS”触摸键实现。



17. 文档数据设置步骤



经由[文档数据]/[文档列表]注册文档数据。

经由[文档数据]/[文档设置]设置[写设置（文档数据发送到 SRAM）]。

当执行自动数据传输时，经由同一对话框设置[PLC&SRAM 直接数据传输设置]，到这里自动传输完成。

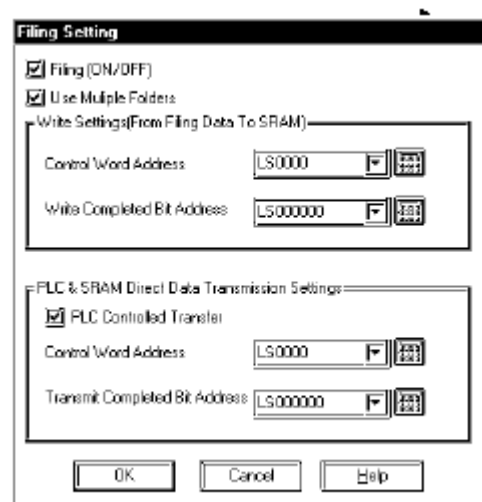
当执行手动数据传输—类型 1 时，设置文档数据显示并指定放置区，到这里手动传输—类型 1 完成。

当执行手动数据传输—类型 2 时，设置库项目并指定放置区。

放置键盘，到这里手动传输—类型 2 完成。

8.3 文档数据设置

对于设置则运用多种文件夹和文档数据触发传输，在选择工程管理器的[屏幕/设置]-[文档数据]-[文档设置]之后，将出现以下的菜单：



1、文档（开/关）

为了运用文档数据选定这个检查框。

2、运用多种文件夹

为了注册 2 个或更多的文件夹选定这个检查框。

3、写设置（从文档数据到 SRAM）

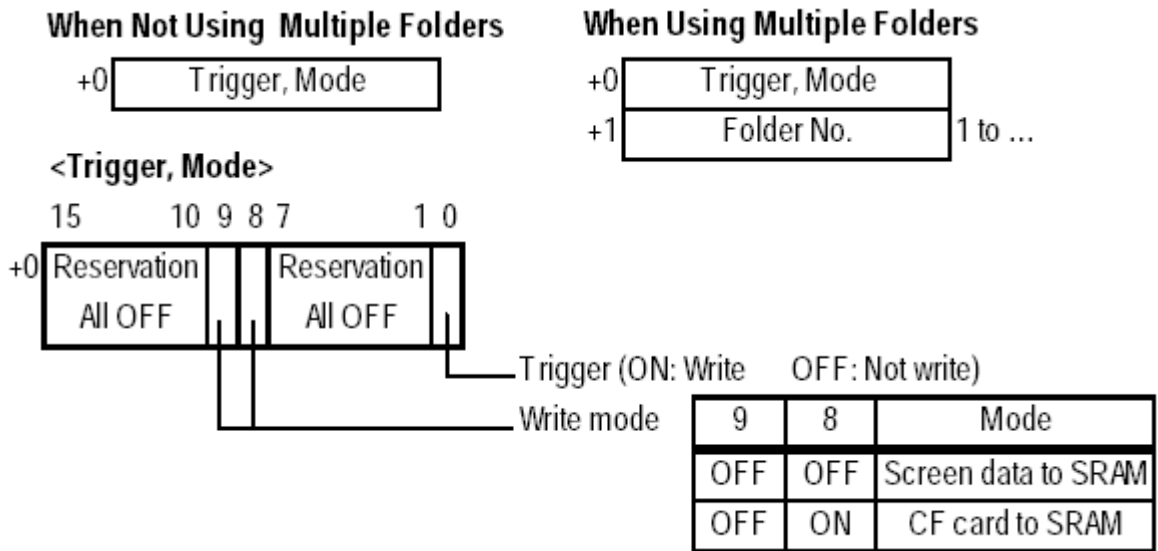
触发响应后，储存在 GP 内部存储器（屏幕数据）或 CF 卡中的文档数据被写入后备 SRAM 中，这是为传输做准备。为了传输数据到 PLC 中，你先需要把数据写到后备 SRAM 中。只有 1 项文档数据能被写入后备 SRAM 中。

当运用文档数据时，要确定“写设置”中的输入数据。

4、控制字地址（从文档数据到 SRAM）

指定一个字地址用于触发，或者当写入后备 SRAM 时，指定使用模式，这个触发字地址将控制写时间。当使用多种文档数据条款时，要指定文件夹号。

控制字地址的确定如下所示：



所允许的文件夹号范围如下所示：

GP 内部存储器：1 到 64。

CF 卡：1 到 8999。

5、写完成位地址（从文档数据到 SRAM）

指定一个特殊的位地址当数据写入 GP 后备 SRAM 完成时它被开启。

如果由于内存不足而使数据不能传输到后备 SRAM 中，字地址 LS2032 的第 9 位将被开启。当再一次把数据传输到后备 SRAM 中时，可利用指定的通讯循环时间或 150ms。无论超过哪一个时间，触发位将被置成“断”状态。

	Write Completed Bit Address	LS2032 Bit 9
Normal data transmission	ON	OFF
Data transmission error	-	ON

6、PLC&SRAM 数据传输设置<SRAM、PLC 之间>

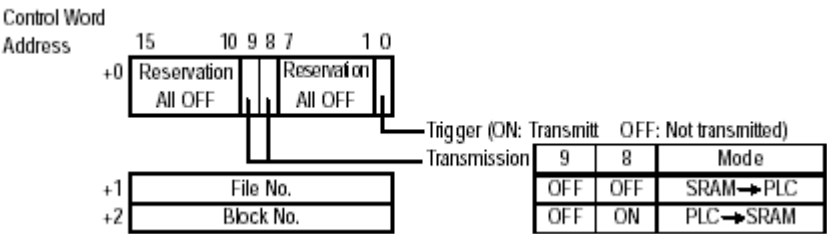
这里，选择用于在后备 SRAM 和 PLC 之间传输文档数据的方法，当利用控制字地址特性执行自动数据传输时，选择这个设置。

7、PLC 控制传输<SRAM、PLC 之间>

当这个检查框被选择时，经由 PLC 触发的自动数据传输被设置，如果这个框没被选定，数据必须在编辑区用工具箱中的文档数据显示来手动传输。

8、控制字地址<SRAM、PLC 之间>

当文档数据传输触发和数据模式被储存后，要求指定一个字地址。



9、传输完成位地址

指定一个位地址当文档数据传输完成时使它开启。

使用二进制数指定文件和模块数。

如果传输不能执行，字地址 LS2032 的第 10 位被开启，为了传输数据到 PLC 中，可利用你的标准通讯循环时间或 150ms。无论超过哪一个时间，触发位将被置成“断”状态。

	Transmit Completed Bit Address	LS2032 Bit 10
Normal data transmission	ON	OFF
Data transmission error	-	ON

8.4 文档数据列表

文档数据的设置可以利用文档数据列表来增加、编辑和删除。在选择了工程管理器中的[屏幕/设置]-[文档数据]-[文档列表]之后，将出现如下菜单：

8.4.1 文档数据列表

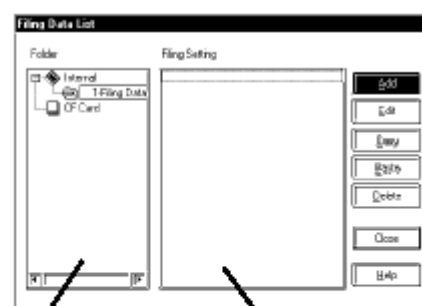
以下是对文档数据设置屏幕的总的看法。

(1)、不使用多种文件夹



The registered
Filing Settings
are displayed as
a list.

(2)、使用多种文件夹



The registered
folders and files
are displayed as
a list.

The registered
Filing Settings
are displayed as
a list.

Add: 增加文件夹和文件设置。

Edit: 编辑文件夹和文件内容。

Copy: 复制文件夹和文件。

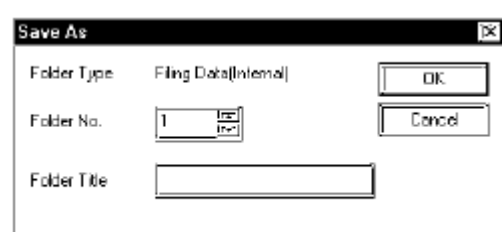
Paste: 粘贴复制的文件夹和文件。

Delete: 删除文件夹和文件。

8.4.2 文档数据注册

注册另外的文档数据设置。

<1>、当使用多种文件夹时，选择文档数据列表对话框并按下[Add]按钮，将出现以下屏幕显示：



<2>、Folder No.（文件夹号）

文件夹号可以被指定在以下范围之内：

内部存储器：1 到 64。

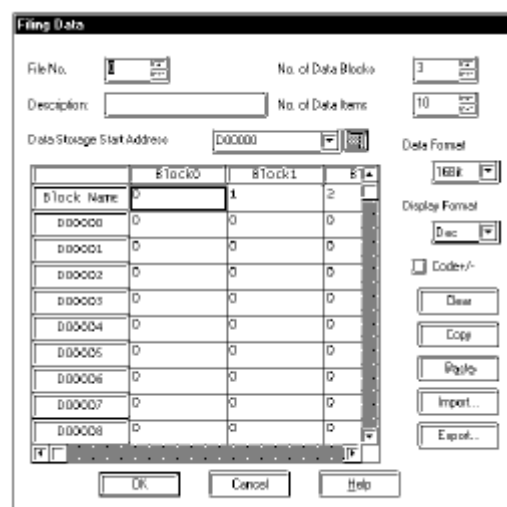
CF 卡：1 到 8999。

<3>、文件夹标题

输入一个文件夹（不允许输入逗号）

<4>、注册文档数据

当使用或者不使用多种文件夹时，选择“1—Filing data”且在文档数据列表对话框中按下[Add]按钮，将出现以下屏幕显示：



(1)、File No. (文件号)

文档数据受约束于单个文件单元。在这里，指定已注册的数据文件号。可以注册不超过 2048 个文件。

(2)、Data Storage Address (文件存储开始地址)

输入存储传输数据的起始地址，数据存储区被连续保存在文档数据条款的指定起始地址中。

(3)、No. of Data Blocks (数据模块号)

指定在一个文件中的模块数量，其最大值为 1650 (条款数量改变其最大值也能改变)。

(4)、No. Of Data Items (数据条款号)

指定在一个模块中的数据条款数量，其最大值为 9999 (模块数量改变其最大值也能改变)。

(5)、Data Format (数据格式)

选择 16 位或 32 位数据。

(6)、Display Format (显示格式)

选择文档数据显示格式。

(7)、Code +/- (+/-代码)

这个检查框被选定后，负显示变成有效。

(8)、Import (输入)

其他文件数据 (CSV 格式) 可以被 输入且用作文档数据。

(9)、Export (输出)

文档数据能被输出且可以用 CSV 格式来保存。

8.5 文档数据设定

输入文件号、描述、数据模块数、数据条款数、数据存储起始地址。

Filing Data

File No.: 0 No. of Data Blocks: 3
 Description: Heat Control No. of Data Items: 10
 Data Storage Start Address: D00100

Block Name	Block0	Block1	Block2
D00000	0	0	0
D00001	0	0	0
D00002	0	0	0
D00003	0	0	0
D00004	0	0	0
D00005	0	0	0
D00006	0	0	0
D00007	0	0	0
D00008	0	0	0
D00009	0	0	0

Data Format: 16Bit
 Display Format: Dec
☐ Code +/-
 Clear
 Copy
 Paste
 Import...
 Export...
 OK Cancel Help

文档数据列表显示

使用多种文件夹

不使用多种文件夹

Filing Data List

Folder: Internal (selected), 1. Settings, CF Card

Filing Setting: 1 Heat Control

Add, Edit, Copy, Paste, Delete, Close, Help

Filing Data List

Filing Setting: Internal Memory (selected), CF Card

1 Heat Control

Add, Edit, Copy, Paste, Delete, Close, Help

8.6 文档数据自动传输

先前输入的文档数据如下所示传输到 PLC 中。这里有三种传输数据的方法：经由 PLC 触发或者从两种有用的屏幕类型中选择所需的设置。在这里，对每种传输方法都做了解释。

Filing Data

File No.1

No. of Data Blocks3

Description:Heat Control

No. of Data Item3

Data Storage Start AddressD00100

Data Format16Bit

	Block0	Block1	Block2
Block Name	~20°C	21~35°C	36°C
D00100	200	30	55
D00101	202	30	56
D00102	205	28	62

Display Format:Dec

Clear

1、文档设置

当控制字地址变化时，模块 1 的数据被传输。

Filing Setting

☒ Filing (ON/OFF)

☐ Use Multiple Folders

Write Settings (From Filing Data To SRAM)

Control Word AddressD00200

Write Completed Bit AddressM00001

PLC to SRAM Direct Data Transmission Settings

☒ PLC Controlled Transfer

Control Word AddressD00201

Transmit Completed Bit AddressM00002

OK

Cancel

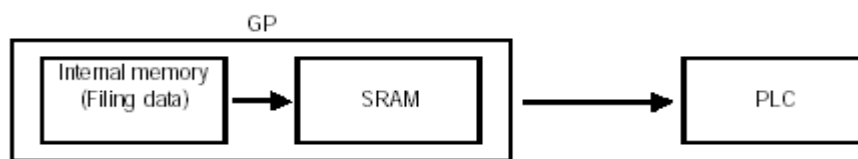
Help

Write Settings (From Filing Data To SRAM)
Control Word Address: D00200
Write Completed Bit Address: M00001

PLC & SRAM Direct Data Transmission Settings
Control Word Address: D00201
Transmit Completed Bit Address: M00002

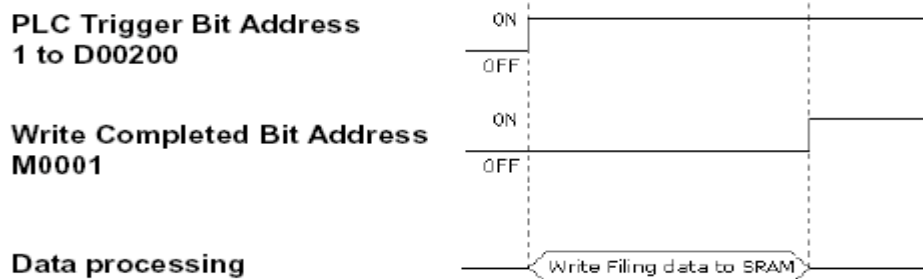
当使用多种文件夹时，控制字地址选择在 D00200 和 D00201 之间。在设置传输控制字地址时，这两个地址被更替使用。

2、文档数据数据传输



<1>、文档数据传到 SRAM

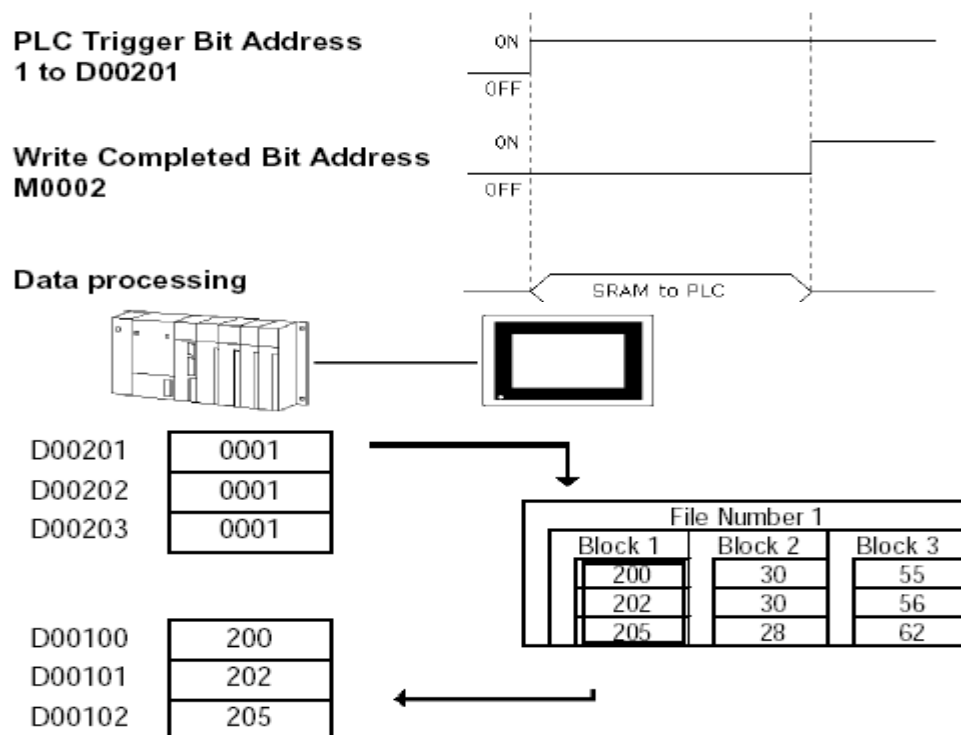
当数据传输控制字地址（D00200）的“0”位被开启时，所有文档数据写到 SRAM 中。



<2>、SRAM 传输到 PLC

当数据传输控制字地址（D00201）的“0”位被开启时，指定的文档数据写入 PLC。

为了指定文档数据，先前传输的文档数据的文件号被存储在 D00202 地址中，而模块号被存储在 D00203 地址中。



8.7 手动文档数据传输实例 1

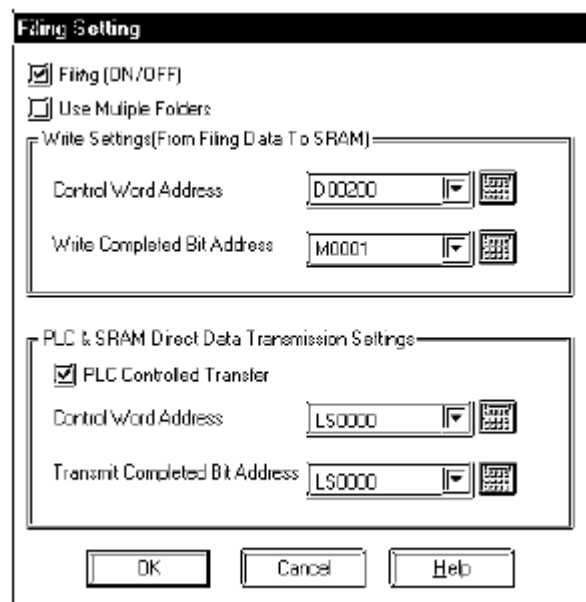
以下是解释如何利用所选屏幕设置来传输数据（手动传输 1）。

1. 屏幕实例

在这个例子中，通过触摸屏幕使数据从 1 号文件模块中传输到 3 号文件中。

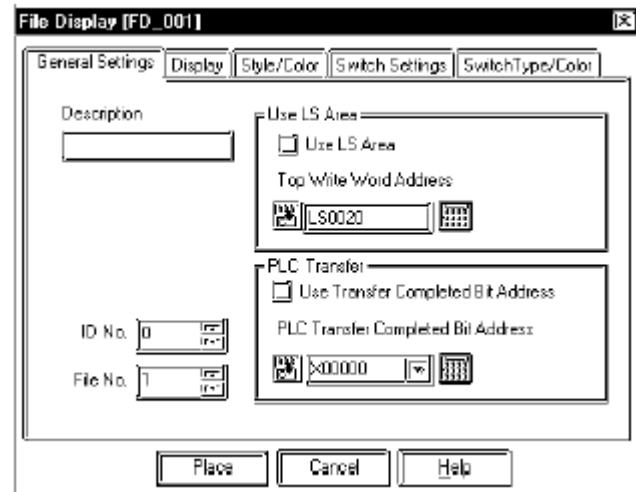
1、文档数据设置实例

使“PLC Controlled Transfer”对话框不被选中，则进入了手动传输模式。



2、文档数据显示设置

当选定手动传输模式后，要放置部品中的文件名显示。



Description (描述): 热控制。

Use LS Area (使用 LS 区): 不选定。在这个实例中，数据从 SRAM 传输到 PLC 中，传输不通过 LS 区，保持默认的设置。

Top Write Word Address (首写入字地址): 不选定。既然这个实例不用到 LS 区，也就不必指定地址，当使用 LS 区时，要指定存储数据的首地址。

PLC Transfer (PLC 传输)

Use Transfer Completed Bit Address (使用传输完成位地址): 不选定。

这个实例中，当数据在 LS 区和 PLC 或者 SRAM 和 PLC 之间传输时，不使 PLC 传输完成字地址开启（当数据传输完成时没有信息被发送）。

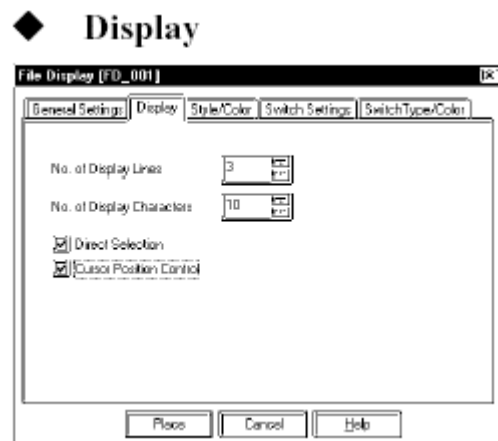
当使用传输完成位地址时，在位开启状态被检测到后，在 PLC 相应的位处于封闭状态。另外，使用传输完成位地址时，数据在 PLC 和 LS 区或者 PLC 和 SRAM 之间没有正常传输完成，则特殊继电器（LS2032）的第 10 位地址将被开启。

PLC Transfer Complete Bit Address (PLC 传输结束位地址): 不指定。

既然这个实例中不使用传输完成位地址，则这个地址不需要被指定。当使用传输完成位地址时，在数据传输完成后指定的地址将被开启。

ID 号 0: 在这个实例中，只有放置一个文档数据显示，则不必输入 ID 号。保持默认设定。当在现有的显示屏上放置两个或多个文档数据显示时，要确定不使用相同的 ID 号。另外，当放置两个或多个文档数据显示在主窗口上时，则要确保不使用相同的 ID 号。

文件号 1: 输入先前输入的文档数据号。



Cursor Postion Control（指针位置控制）

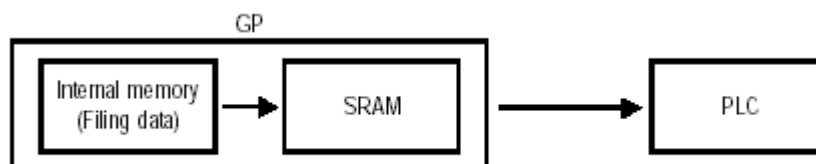
即使在 GP 监控器上的屏幕有变化，当前的文件名显示的指针地址能被保存。

当设置文件名显示时，指针位置控制能被指定。

当 GP 的主能量被开启或者 GP 被复位时，指针将显示在第一行。

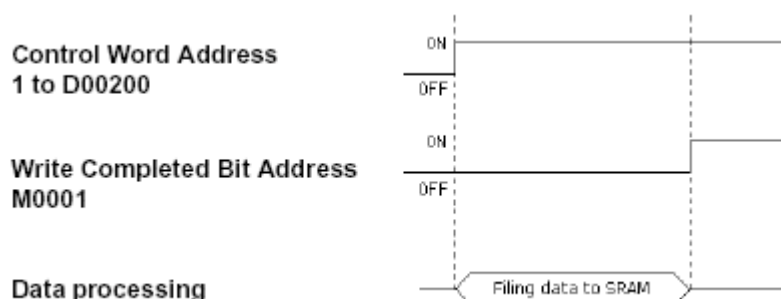
指针位置将被保存在每一个 ID 号中，为了保存指针位置，要确定文档数据显示的 ID 号无误。

4. 文档数据传输流程



（1）、文档数据传输到 SRAM

当触发键触发时，所有的文档数据被写入 SRAM 中。

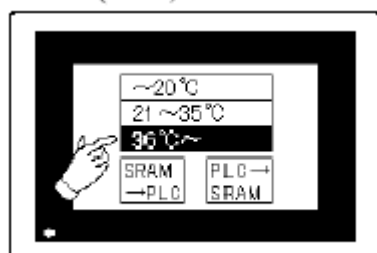


（2）. SRAM 传输到 PLC

通过 GP 上的触摸键使选定的文档数据写入 PLC 中。

5. 操作手续

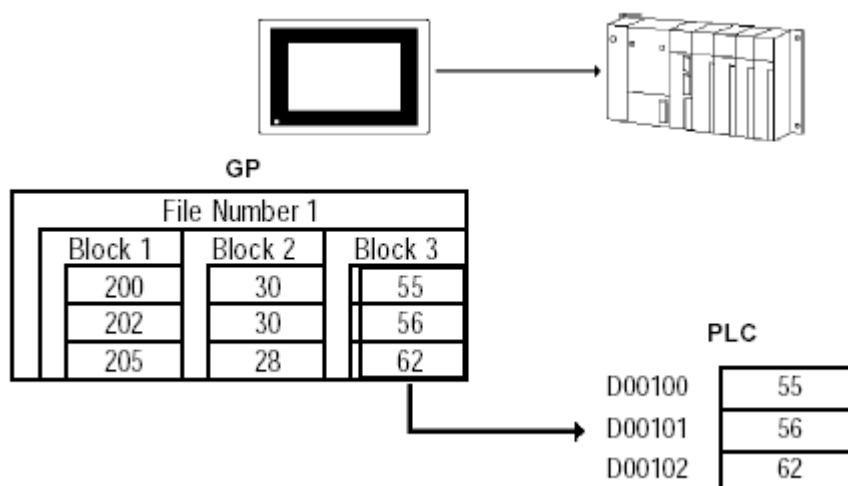
1)、选择一个条款（这里，选择模块 3）。



2)、按“SRAM 传输到 PLC” 按键。



3)、文档数据从 SRAM 传输到 PLC 中。

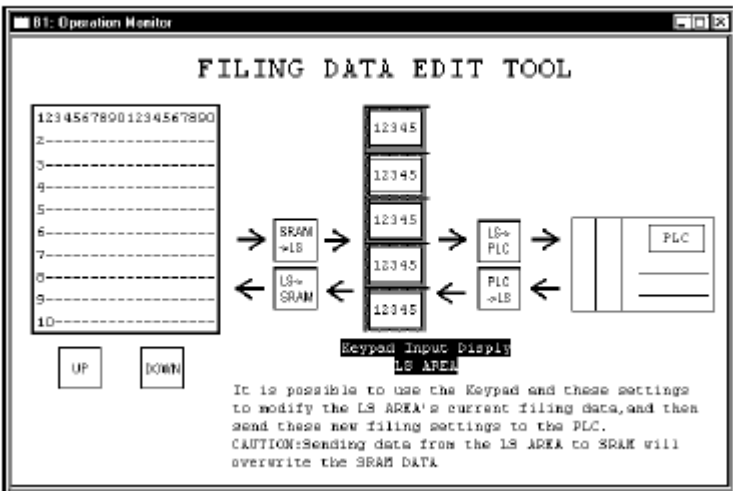


8.8 手动文档数据传输实例 2

这里，在屏幕上利用文件名显示来选择一个项目并通过 LS 区调整数据，然后传输数据。

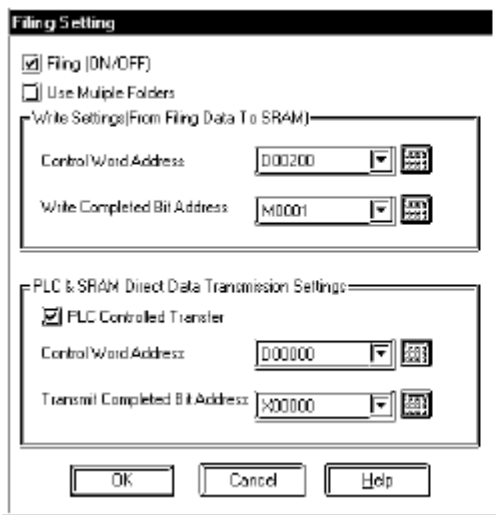
1、屏幕实例

■ Screen Example



2、文档设置实例

当这个对话框没被选定时，手动数据传输将被执行。



Write Setting (From Filing Data To SRAM)

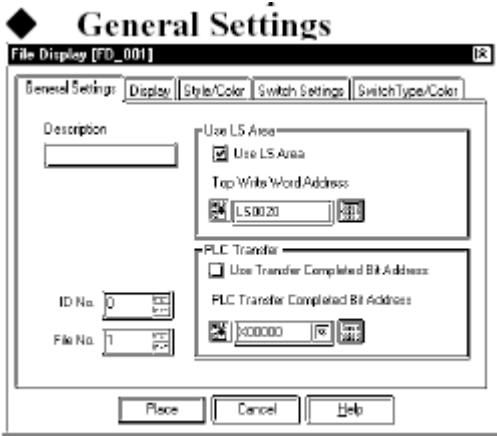
Control Word Address: D00200
Write Completed Bit Address: M00001

PLC & SRAM Direct Data Transmission Settings

PLC Controlled Transfer: Not selected

3、文档显示设置

当手动传输模式选定后，在部件中应选择“File Data Display”放置。

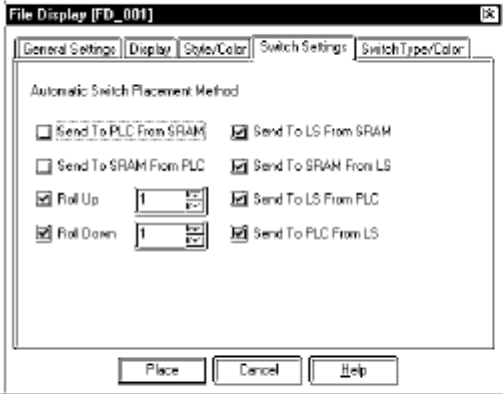


Use LS Area: 选定。文档数据在被传输到 PLC 之前，首先被传输到 GP 的 LS 区。当文档数据在 SRAM 和 LS 或者 LS 和 SRAM 之间传输时，GP 的内部特殊继电器（LS2032）的 11 位将被开启。

Use Transfer Completed Bit Address（利用传输完成位地址）：不选定。

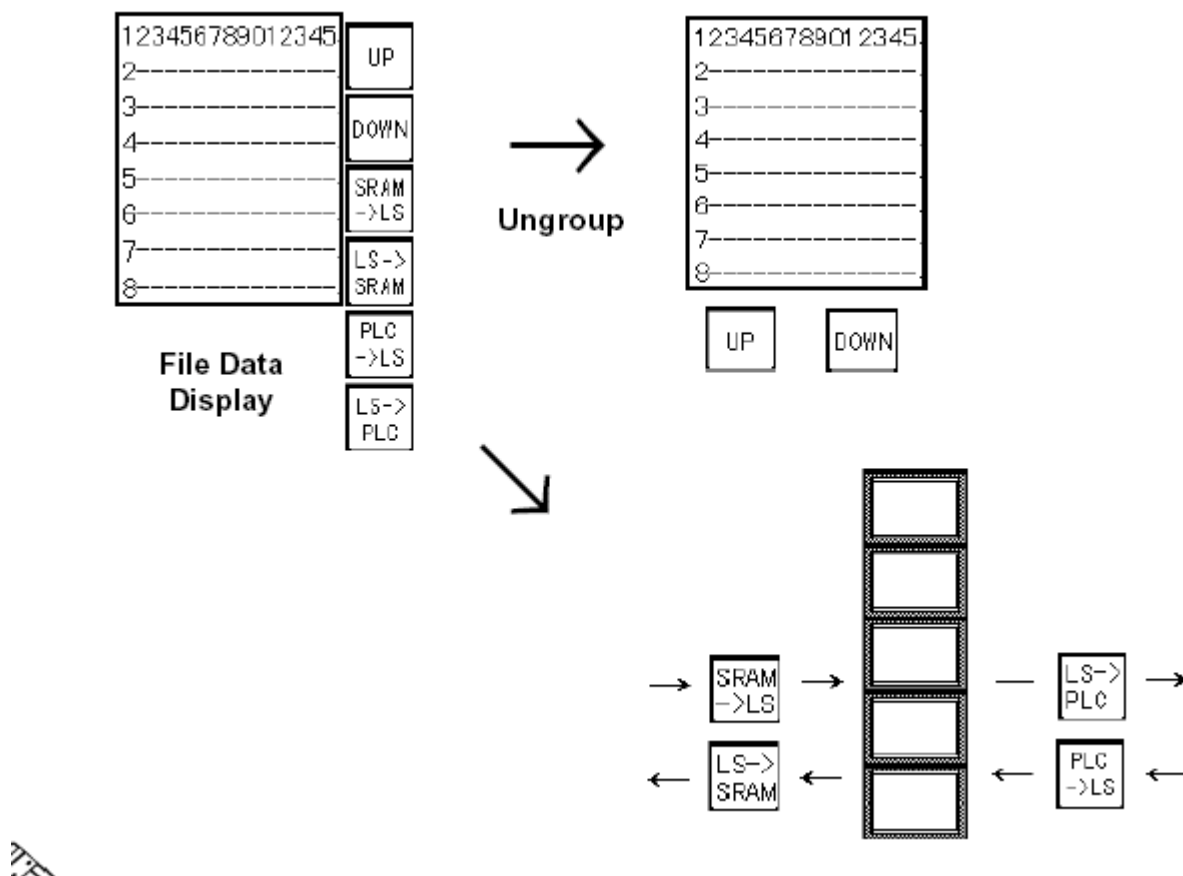
ID NO.:0。在这个实例中，只放置一个文档数据显示时，不必输入 ID 号，保持默认设置。当放置两个或更多文档数据显示时，要指定文件号。

◆ Switch Settings



- Send To PLC From SRAM: Not Selected
- Send To SRAM From PLC: Not Selected
- Roll Up: Selected
- Roll Down: Selected
- Send To LS From SRAM: Selected
- Send To SRAM From LS: Selected
- Send To LS From PLC: Selected
- Sending To PLC From LS: Selected

放置：按下放置键显示文档数据显示。如果你想改变部件的位置或属性，你需要把聚集起来的部件先取消组，并把需要的放置在所需位置。



4、实际操作过程

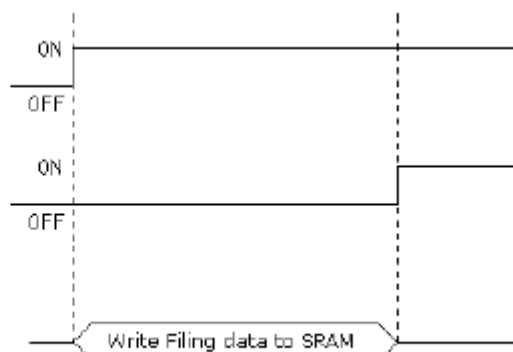
1)、写文档数据到 SRAM

当触发键被触发时，所有的文档数据被传输到 SRAM 中。

Trigger Bit Address
D0020000

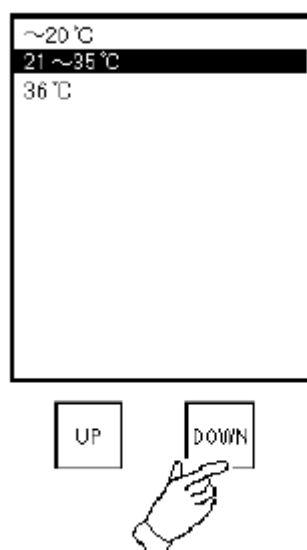
Write Completed Bit Address
M0001

Data processing



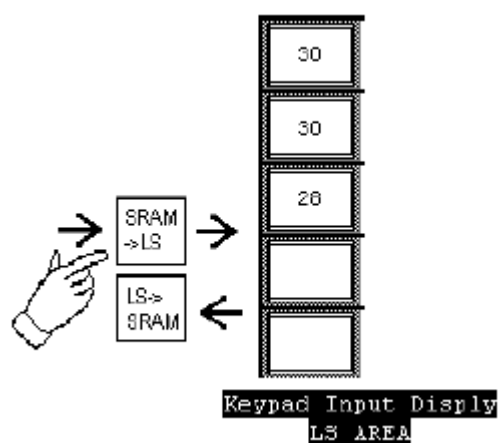
2)、选择文件名

通过触摸 GP 上的模块 2，选择所需设置。



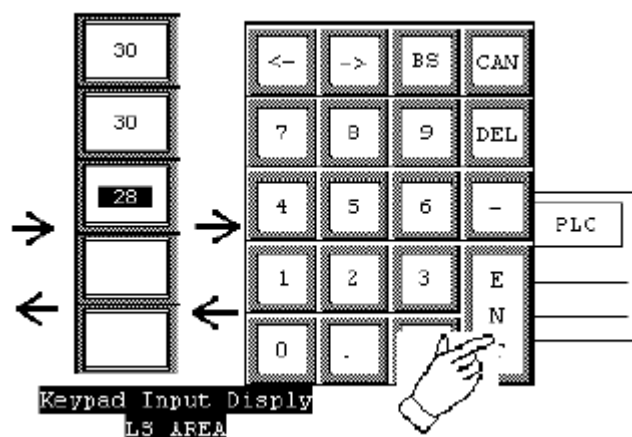
3)、后备 SRAM 传输到 LS 区

触摸“SRAM→LS”按键使数据从 SRAM 传输到 LS 区。



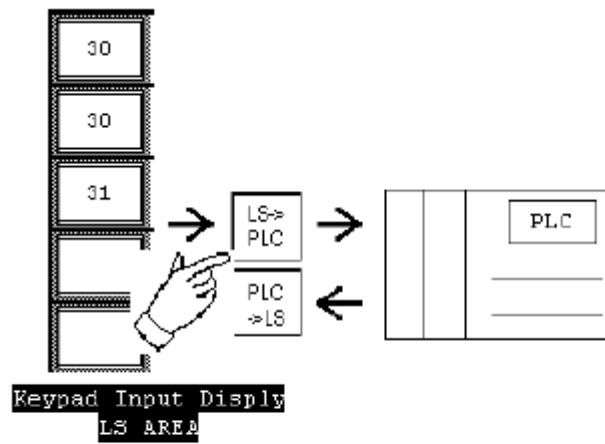
4)、数据编辑

触摸键盘输入显示工具并弹出键盘。利用弹出键盘编辑数据。



5)、LS 区传输到 PLC

触摸“LS→PLC”按键使数据从 LS 区传输到 PLC。



9 扩展串口的应用

这个功能仅适用于 GP2000 系列（除 GP2301，2501，2401H 和 2301H 外）
扩展串口通过 D 脚本的编写能够方便的实现通讯（RS232 口），下面是 D 脚本的编写语句的说明：

9.1 接收功能

下面的命令被用于读从扩展口接收到的数据。

格式

IO_READ([p:EXT_SIO],LS Storage Address,Transer Byte)

例如

IO_READ([p:EXT_SIO], [w:LS0100], 10)

[p:EXT_SIO]表示接收数据

[w:LS0100]这是存放接收数据的起始地址，接收数据的大小存放在 LS0100，接收的数据从 LS101 开始存，10 表示有 10 个字节存放在 LS 区（最大传送字节是 6095）。

9.2 发送功能

下面的命令表示把要发送的数据写到扩展口

格式 IO_WRITE([p:EXT_SIO],LS Storage Address,Transer Byte)

例子 IO_WRITE([p:EXT_SIO], [w:LS0100], 10)

[w:LS0100]是要发送的数据的开始地址，10 表示要发送 10 个数据，

*注意：每一个传送的是字节，也就是说传送的是每一个字的低八位，同样接收的数据也是一样的，如下图所示

LS0100	00	Byte 1
LS0101	00	Byte 2
LS0102	00	Byte 3
LS0103	00	Byte 4
LS0104	00	Byte 5
LS0105	00	Byte 6
LS0106	00	Byte 7
LS0107	00	Byte 8
LS0108	00	Byte 9
LS0109	00	Byte 10

9.3 控制

增加控制变量去清除发送和接收缓存器以及错误状态，这个变量的属性是只能写的。

格式

位 [c:EXT_SIO_CTRL**]**:00 to 15

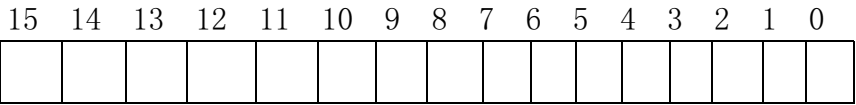
字 [c:EXT_SIO_CTRL]

例如：控制字的内容

位 [c:EXT_SIO_CTRL00]=1

字 [c:EXT_SIO_CTRL]=0x0007

当控制字的内容被确定以后，它的处理过程按照这样的次序来完成的，先清除错误状态，然后清除接收缓存，再清除发送缓存。



位	内容
15	保留
14	保留
13	保留 e
12	保留
11	保留
10	保留
9	保留
8	保留
7	保留
6	保留
5	保留
4	保留
3	保留
2	1:清除错误状态
1	1:清除接收缓存
0	1:清除发送缓存

9.4 状态

增加[EXT_eSIO_ISTAT]作为状态变量，这个变量的属性是只读的。

格式

位 [s:EXT_SIO_STAT**]**:00 to 15

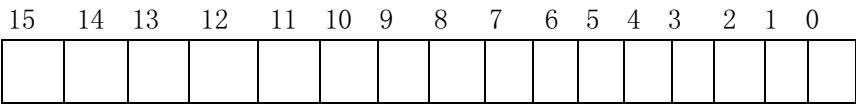
字 [s:EXT_SIO_STAT]

例如

位 if([s:EXT_SIO_STAT00]=1)

字 if(([s:EXT_SIO_STAT]&0x0000)<>0)

字的内容描述如下图



位	内容
15	0:没有扩展 SIO 协议 1:有扩展 SIO 协议
14	保留
13	保留
12	保留
11	保留
10	保留
9	保留
8	保留
7	保留
6	保留
5	保留
4	保留
3	0:正常 1:接受错误
2	0:没有接受到数据 1:接受到数据
1	0:正常 1:发送错误
0	0:发送缓存有数据 r 1:发送缓存没有数据

注意：1) 只有需要检测的位去检测，保留的位留作以后用。
2) 当数据传送发生错误时，数据传送错误的位便置 1。
3) 当数据接受错误发生时，数据接收错误位便置 1。

9.5 接收数据的大小

这个属性也是只读的。

格式[r:EXT_SIO_RCV]

例如 if ([r:EXT_SIO_RCV]>=10)

9.6 扩展通讯口的设置

这个设置可以在离线方式（OFFLINE）或者在 GP system 中都可以。

设置	设置范围	设置值
传送波特率	2400,4800,9600,19200,38400(bps)	9600 bps
数据长度	7,8(bits)	8 bits
停止位	1,2(bits)	1 bits
校验位	None ,Odd ,Even	None

9.7 通讯举例

现在有一个通讯仪表，它有一个 RS—485 口，它支持多种波特率。首先在 GP 的扩展 SIO 设置中选择和仪表的通讯设置一样的参数。（一个起始位，8 位数据，无校验位，一个或两个停止位）

新建一个工程，PLC 类型选择 Memory Link SIO type, 协议选择 General SIO protocol, 然后新建主画面，在主画面上建立如下几个部件：

- 1) 位开关 它的地址是[b:005200], 相对应的符号是初始化，功能是”momentary”
- 2) 位开关 它的地址是[b:005200], 相对应的符号是自动收发，功能是 “bit set”
- 3) 再可以放若干个用于输入要发送的数据 “数据输入” 部件和相对应的要显接收数据用的 “数据显示” 部件，这样就可以很清楚的看到通讯是否成功了。

这样再在 D 脚本编写程序，可以参考如下：

第一个程序：触发条件是[b:005200]的上升沿

```

if ([s:EXT_SIO_STAT15]==1)      //判断端口的状态，检查是否有扩展 SIO 协议
{
[w:通讯步骤]=1
[w:超时设定]=3                //3 秒的超时（这里的数据可以改变）
[w:下一通讯]=1
[c:EXT_SIO_CTRL]=0x7          //将发送缓存和接收缓存区都清 0
}
endif

```

第二个程序是主程序，触发条件是[b:203802]上升沿

```

if ( 1 )
{
if ( [b:通讯允许]==1 )      //允许通讯

```

```

{
    if ( [w:下一通讯]==1 )          //有多个通讯命令时，用此控制
    {
        if ( [w:通讯步骤]==1 )      //第 1 步 发送
        {
            if ( [s:EXT_SIO_STAT00]==1 )    //发送缓冲为空
            {
                IO_WRITE([p:EXT_SIO], [w:0080], 8) //发送数据
                [w:通讯步骤]=2                //发送完成标志
                [w:接收超时]=[w:超时设定]     //接收超时
            }
        }
        endif
    }
}

endif          //END 通讯第一步
if ( [w:通讯步骤]==2 )          //第 2 步 接收数据
{
    if ( [s:EXT_SIO_STAT02]==1 )    //接收缓冲区有数据
    {
        if ( [r:EXT_SIO_RCV]>=10 )    //接收 10 个数据
        {
            IO_READ([p:EXT_SIO], [w:0100], 10)
            [w:通讯步骤] = 10          //完成读数据
            [w:下一通讯]=1              //多个通讯命令，改此值
            [b:通讯允许]=0
        }
    }
    endif
}

endif

if ( [w:接收超时]==0 )          //发生超时
{
    [w:通讯步骤] = 11
    [w:下一通讯]=1                //重复发送该命令
    [b:通讯允许]=0
}

endif

}

endif          //END 通讯第二步
}

endif

}

endif

endif

```

第三个程序：它用于检测通讯是否超时，触发条件是每隔一秒运行一次

```

if ([w:接收超时]>0)
{ [w:接收超时]=[w:接收超时] - 1
}
endif

```


10 Memory-link 通讯

10.1 关于 Memory-link

GP 的通常应用方法是与 PLC、变频器、伺服器、温度控制器等进行通讯，这些应用都不需要编写通讯程序，直接在画面里需要的地方指定 PLC 地址即可。我们称为“Direct Access (Programless)”、无需编程的直接访问方式。这种方式对应了各厂家的 500 种以上的控制器型号。

Memory-link 方式则可用于 1:1 或者 1:n 的 GP 与非通用设备的通讯，例如与个人计算机、单片机系统。

10.2 通讯基础

站:

站是通讯系统中的一个单元或设备，它们可以是 GP、PLC 的 CPU、PLC 的通讯处理模块、带通讯口的个人计算机。

硬件接口和传送媒介:

串行通讯采用标准的 RS-232、RS-422 或者 RS-485 接口。低速度通讯时，可以采用非屏蔽双绞线、屏蔽双绞线等。

访问方式:

主从方式通讯：这是简单有效的通讯方式。

在大多数主从方式通讯中，主站控制通讯量。主站把数据发送至连接的从站，从站根据提示返回或不返回数据。

在大多数主从方式通讯中，从站之间不直接通讯。

大多数 Direct Access (Programless)情况下，GP 与控制器的通讯，就是主从方式，并且 GP 是主站、控制器是从站。当然也有例外。

但是 Memory-link 方式下，GP 与设备的通讯，也是主从方式，但 GP 是从站、对方设备是主站。

协议:

协议是通讯伙伴之间为了执行指定的通讯服务而规定的精确数据定义。它定义了通讯数据的相关目录结构。

ASCII、BIN 传送:

这是通讯协议中每个字节数据的传送方式。

DTR(ER)、XON/OFF 控制:

这是两种在通讯繁忙时防止缓冲区数据溢出的控制方式。

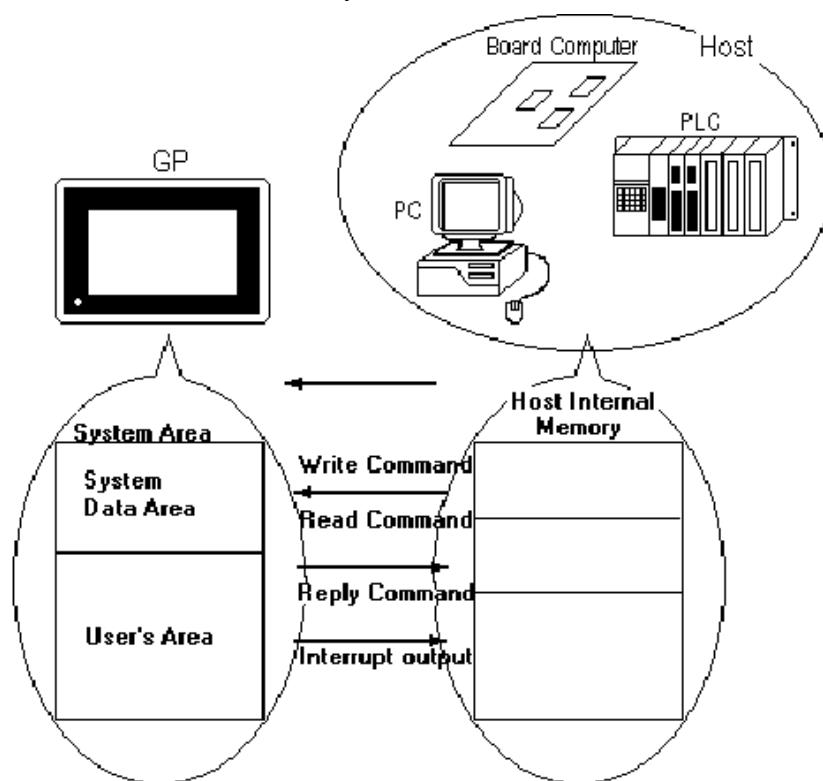
10.3 Memory-link 通讯基础

大多数 Direct Access (Programless)情况下, GP 与控制器的通讯, 就是主从方式, 并且 GP 是主站、控制器是从站。当然也有例外。

但是 Memory-link 方式下, GP 与设备的通讯, 也是主从方式, 但 GP 是从站、对方设备是主站。

在 Direct Access (Programless)方式下, 是不需要编写通讯程序的, 直接指定需要的控制器数据地址即可。GP 系统程序根据这些地址设定和控制器的协议, 来完成通讯控制。

但是 Memory-link 方式下, GP 是从站, 根据前节访问方式的说明, 因此 GP 是根据主站(Host)传来的数据提示进行工作, 例如显示主站(Host)传来的数据、发送数据到主站(Host)。主站(Host)侧, 需要依任务编写程序, 例如主站需要和 GP 进行哪些数据通讯、通讯协议(数据格式)则是依据 Memory-link 的规定。



10.3.1 Memory-link 通讯时 GP 内部数据区

GP 内部数据区范围是 0-4095 个 WORD。除了 GP 内部系统数据区, 都是用户数据区。

GP 内部系统数据区, 是 GP 运行的系统参数、状态等的特殊存储器区域。其每个地址内容都是预定义好的。

LS0000-LS0019
LS0020-LS2031
LS2032-LS2047
LS2048-LS2095
LS2096-...

系统数据区
用户数据
特殊继电器区
保留
用户数据

写入数据可以控制画面、错误信息等

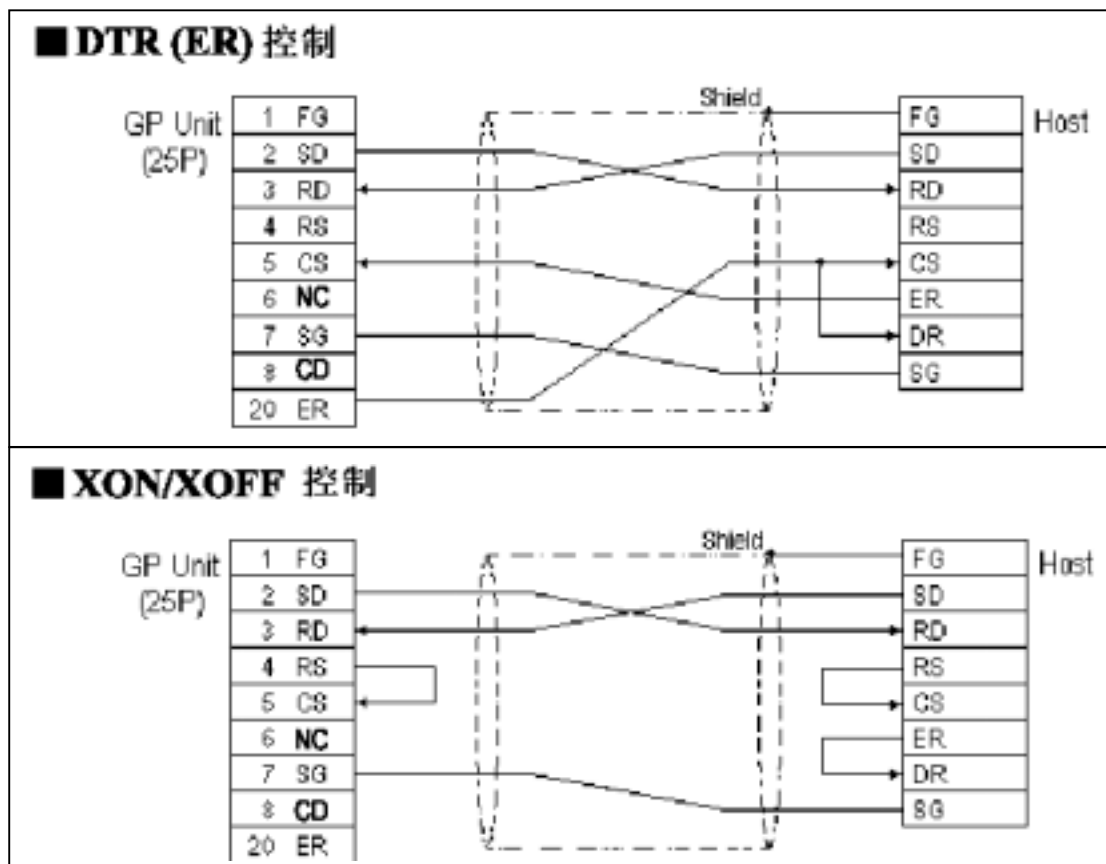
GP 的状态信息

10.3.2Memory-link 通讯时 GP 内部系统数据区

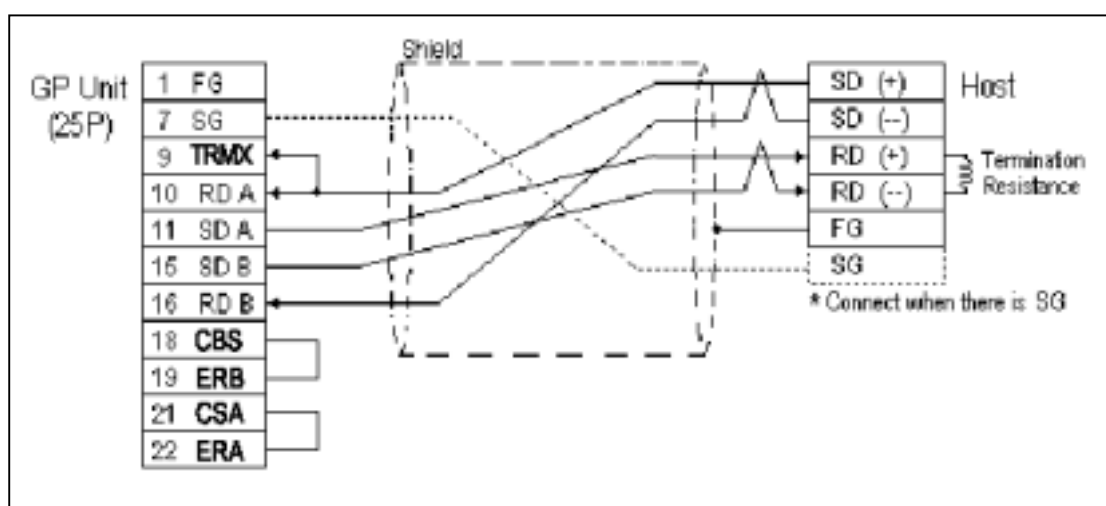
地址	内容	功能	bit	解释
1	状态		0,1	保留
			2	正在打印
			3	数据写入
			4-7	保留
			8	K-tag 输入出错
			9	显示 0:on/1:off
			10	背景灯故障检测
			11	触摸板输入错误
			12-15	保留
3	错误状态		0,1	保留
			2	系统 RAM/ROM
			3	画面检查和
			4	SIO 帧
			5	SIO 校验
			6	SIO 溢出
			7,8	保留
			9	内部存储器需要初始化
			10	时钟
			11-15	保留
4-8	时钟	BCD2 位		年、月、日、时、分
10	中断输出	触摸 OFF		触摸 OFF 后低 8bit 作为中断输出
11	控制		0	背景灯
			1	峰鸣器
			2	打印
			3	保留
			4	峰鸣器 0 允许/1 禁止
			5	AUX 输出 0 允许/1 禁止
			6	触摸使显示转为 ON 时, 0:禁止中断输出 /1:允许
			7	保留
			8	VGA 显示, 0 禁止/1 允许
			9, 10	保留
			11	硬拷贝数据, 0 允许/1 禁止
			12-15	保留
12	屏幕显示			FFFF: 立即清除/0: 显示
13	中断输出			低 8bit 作为中断输出
15	显示画面号	当前或改变		1-8999(BCD 格式时 1-1999)
16	窗口控制		0	0: 不显示/1: 显示
			1	0: 可改变叠放顺序/1: 不可
17	窗口号			注册的窗口号
18-19	窗口位置			X、Y 坐标

10.3.3 Memory-link 通讯时电缆连接

RS-232 在 DTR (ER) 控制、XON/XOFF 控制方式时的电缆连接。这是两种在通讯繁忙时防止缓冲区数据溢出的控制方式。



RS-422 的电缆连接



10.4 Memory-link 通讯协议

通讯协议规定了 GP 和主机(Host)之间数据通讯的格式。根据应用需要和数据处理能力，Memory-link 通讯协议有以下几种工作方式。

Memory-link	ASCII 码兼容方式(Compatible)	1:1	ASCII 码
	扩展方式(Extended)	1:1	ASCII 码
			BIN 码
		1:n	ASCII 码
			BIN 码

■ ASCII 码兼容方式(Compatible)

在这种方式，只有写到数据区的命令(ESC W)和读数据区的命令(ESC R)。这是基本的 ASCII 码交换协议。因此，在主机(HOST)控制器控制通讯的负担（程序）比较小。相对地，传送和接收数据的灵活性降低。

■ 扩展方式(Extended)

在这种方式，除了写到数据区的命令(ESC W)和读数据区的命令(ESC R)，还可以执行绘图命令。它是一种支持 GP 和控制器多点连接的通讯协议。为了提高数据通讯的灵活性，可以指定检查和（SUM CHECK）、ACK、NCK 是否使用。使用 ASCII 或 BIN 码，则依据控制器的软件开发环境。

10.4.1 ASCII 码兼容方式(Compatible)通讯命令

GP 与主机(HOST)之间有四条通讯命令。

主机(HOST)读命令： 主机(HOST)读 GP 的数据

Esc	R	0	0	6	4	0	0	0	2	CR
-----	---	---	---	---	---	---	---	---	---	----

读命令

读数据长度占4字节,例:读2字

起始

读的地址占4字节,例:地址100

结束

其中：

读的数据起始地址占 4 字节，例如读地址 100，其 HEX 格式为 0064HEX。

读的数据长度也占 4 字节，数据长度是以字为单位的。

GP 应答命令： GP 响应主机(HOST)的提示、并返回数据

Esc	A	每1字的数据:用4字节传送	每1字的数据:用4字节传送	CR
-----	---	---------------	---------------	----

起始

应答

返回第1字的数据

返回第2字的数据

结束

其中：

每个地址的（WORD）数据，占 4 字节传送。

主机(HOST)写命令：主机(HOST)写数据 GP

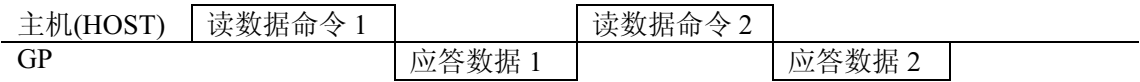
Esc	W	0	0	6	4	每1字的数据:用4字节传送	每1字的数据:用4字节传送	CR
写命令		写入地址占4字节,例:地址100				写入第1字的数据		写入第2字的数据
起始								

其中:
写入数据起始地址占 4 字节, 例如写地址 100, 其 HEX 格式为 0064HEX。
写的数据每个地址占 4 字节。

GP 中断输出命令：GP 主动发送一个字节数据到主机(HOST)。在 GP 侧系数数据区地址 13 里, 写入一个字节数据 (范围 20H-FEH), GP 主动将此代码发送到主机(HOST)。

10.4.2 ASCII 码兼容方式(Compatible)通讯举例

应用：主机(HOST)连续读 GP 的数据



```
最简单的 QBASIC 程序：
CLS
REM baud, hardware handshaking is ignored and buffers are enlarged.
OPEN "COM1:9600,N,8,1,CD0,CS0,DS0,OP0,RS,TB2048,RB2048" FOR RANDOM AS #1
FOR I = 0 TO 9
  REM Write I value to LS0100
  A$ = CHR$(27) + "W" + "0064" + "000" + CHR$(48 + I) + CHR$(13)
  PRINT #1, A$
  REM Read LS0100 – LS0101
  A$ = CHR$(27) + "R" + "0064" + "0002" + CHR$(13)
  PRINT #1, A$
  INPUT #1, DATA$
  PRINT DATA$
NEXT
CLOSE #1
```

10.4.3 扩展方式(Extended)通讯命令

扩展方式(Extended)的通讯命令，请参考专门的协议说明手册。

11 报警显示

11.1 A-tag 当前报警显示

11.1.1 关于 A-TAG

- 每个警报讯息(文字资料), 登录在 **Text Screen**.(文字画面)
- 监控的 **Bit** 位置, 对应到 **Text Screen** 的行数. 设定的监控的 **Bit** 在 **Word** 地址里.
- 只显示对应 **Bit ON** 的警报文字, 当 **Bit OFF** 时, 显示消失.
- 在一个 **Text Screen**(文字画面)里, 最多可输入 **512** 行文字. 所以最多可显示 **512** 行警报讯息. 但是大多数的 **GP**, 一个画面仅可显示 **40** 行*1 讯息. 如果讯息数超过一个画面时, 建立多个以 **A-tag** 设定的 **Base Screen**.使用切换画面的方式, 显示其它讯息. 就这个方式而言, 设定每个画面的起始行数(**Start Line**), 来延续前一个画面的显示.
- 每行最多可显示 **80** 个字(或 **40** 个中文字). 实际因 **GP** 机种而有所不同.
- 文字的大小, 颜色及框线, 是可设定的选项
- 使用对应每个讯息 **Sub-Screen**(说明画面)功能, 亦可显示. 可与 **Window Display**(窗口显示)组合使用. 说明画面的显示与其它建立在 **Base Screen** 或 **Window Screen** 的 **Tag**(如 **L-tag**)结合动作. 有三种显示

-**Change Display** 切换显示到其它画面

-**Library Display(Base Screen)** 可在窗口中显示资料及图形

-**Text Display(Text Screen)** 在长串说明文字工作下最佳

- 使用作 **Sub-Screen**(说明画面)的 **Base Screen** 或 **Text Screen** 的画面号码,必须是连续的号码及符合文字资料的显示顺序.
- 当使用 **Sub-Screen**(说明画面), 必须设定选择文字行的屏幕按键. (在 **T-tag** 选项的 **A-tag** 选择键)

Sub-Screen(说明画面)以间接方式呼叫, 指定一个 **Word** 地址, 用 **Sub-Screen**(说明画面)显示资料的储存地址.

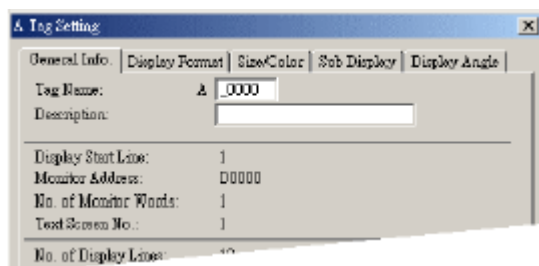
- 当监控的 **Bit** 对应的文字行是空白的(**Blank Line**), **A-tag** 允许删除(隐藏)这一行. 所以 **A-tag** 可以只显示包含警报文字的行数, 无关于空白行的显示特性是 **ON** 或 **OFF**. 最多 **512** 行警报消息正文可被显示. 这个特性仅支持 **GP377**, **GP77R**, **GP37W2** 及 **GP2000** 系列机种.

11.1.2 使用 A-tag

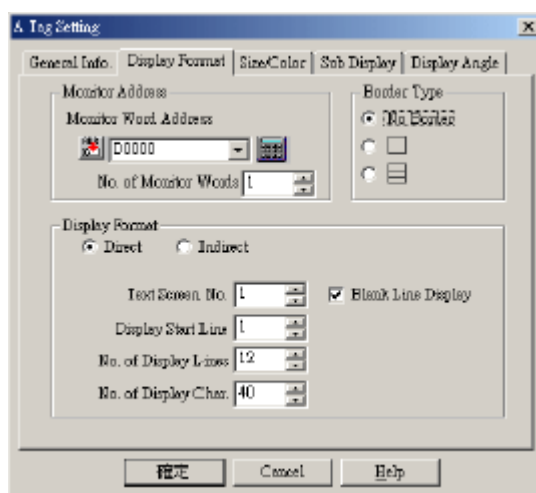
- General Info – 一般信息,提供 A-tag 目前设定的相关信息.

Tag Name - 卷标名称,须小于等于 5 个英文或 2 个汉字字符.

Description - 叙述,须小于等于 20 个英文字符或 10 个汉字字符.




- Display Format – 显示格式



Monitor Address - 监控地址

Monitor Word Address - 监控地址,定义起始的监控 **Word** 地址.

No of Monitor Words – 监控 **Word** 数,定义共几个 **Word** 使用为监控的 **Bit**, 设定依据登录在 **Text Screen** 的行数,最多可输入 32 个 **Words**.

Border Type – 框线型式,可定义的框线型式为***No Border**(无框线)*, *□外框线*及*表格*

Display Format – 显示格式

Direct – 直接,直接定义 **Text Screen** 的号码.

Text Screen No. - 文字画面的号码,定义显示的文字画面的号码.

Display Start Line – 显示的起始行数,定义当监控 **Bit ON** 时,从 **Text Screen** 的那一行显示.

No. of Display Line – 显示的行数,设定一个画面,显示的行数. 最多可设定 40 行.

No. of Display Char. – 每行的字数,设定每行最多可显示几个字,最多 80 个字.(中文字 40 个)

Blank Line Display – 空白的文字行,当监控的 **Bit** 对应的行数是空白时,**A-tag** 允许隐藏此行. 仅显示有文字的行数. 不管 **Blank Line Display** 是 **ON** 或 **OFF**, 最多 **512** 行可显示. **Blank Line Display** 的功能仅提供给 **GP377,GP77R** 及 **GP2000** 系列机种.

Display Format – 显示格式

Indirect – 间接,以间接方式指定 **Text Screen** 号码.本功能仅提供给 **GP377, GP77R** 及 **GP2000** 系列机种.

Text Screen No. Word Address – 储存文字画面号码的 **Word Address**,指定储存文字画面号码的 **Word Address**, 画面号码以二进制格式储存, 当改变文字画面的号码时, 请确认文字画面的行数是相同的.

当 **Text Screen** 的行数, 超出一个画面显示时, **Start Line**(起始行数)的规划如下:

画面 1, **Start Line** = 1

画面 2, **Start Line** = 画面的显示行数+ 1

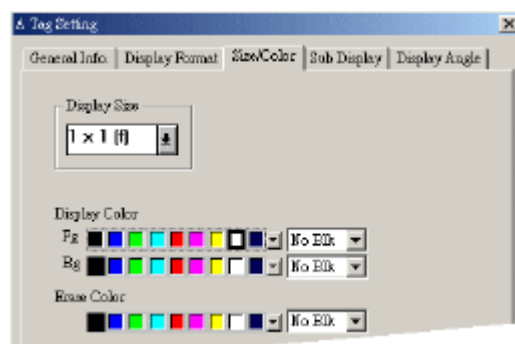
画面 n, **Start Line** = 画面的显示行数 × (n-1) + 1

■ **Style** – 尺寸/型式

Display Size – 显示大小,指定显示的字符大小.

Display Color – 显示颜色,定义前景(**Fg**), 背景(**Bg**)及闪烁(**Blk**)的颜色属性.

Erase Color – 清除颜色,定义未显示区域的颜色属性.单色机种, 请指定黑色.



■ **Sub Display** - 警报说明画面

None – 关闭功能,关闭警报说明画面功能.

Base Screen – 基本画面,执行警报说明画面功能.

Active Window – 动态窗口,执行警报说明画面及窗口功能.

◆ **Sub Display/ Base Screen** – 警报说明画面/基本画面

Base Screen – 基本画面

Mode Settings – 模式设定

Change Screen – 改变画面,改变显示画面到另一个画面, 如同一般换页动作.

Library Display – Library 显示,以 **L-tag** 设定警报说明画面显示位置.

Text Display – 文字显示,以 **X-tag** 设定警报文字说明画面显示位置.

Change Screen – 改变画面

Screen Type – 画面型式

Direct – 直接,直接指定在画面上显示警报说明画面的 **Base Screen** 号码.

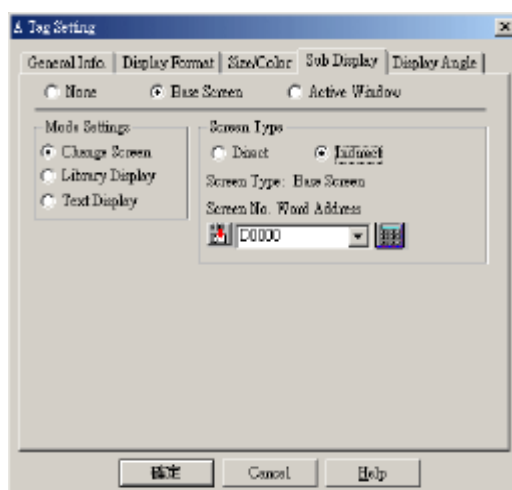
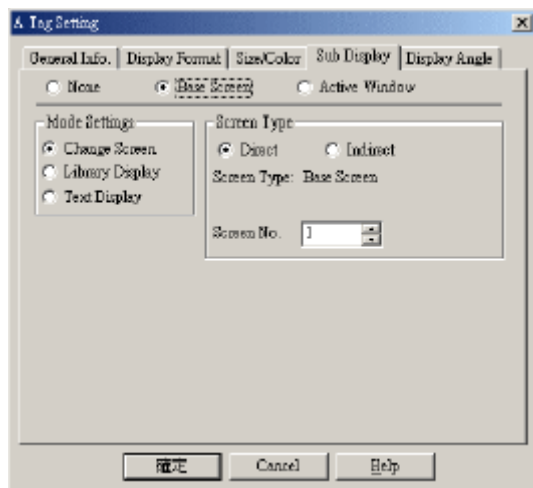
Screen No. - 画面的号码,输入显示警报说明画面的 **Base Screen** 号码. 必须是二

进制格式.

Screen Type – 画面型式

Indirect – 间接,间接指定在画面上显示警报说明画面的 **Base Screen** 号码.

Screen No. Word Address – 储存画面,号码的 **Word** 地址设定储存画面号码 **Word** 地址.**Word** 地址的值, 必须是二进制格式.



Library Display

Screen Type – 画面型式

Direct – 直接,直接指定在画面上显示警报说明画面的 **Base Screen** 号码.

Screen No. - 画面的号码,输入显示警报说明画面(**Base Screen**)的起始号码. .

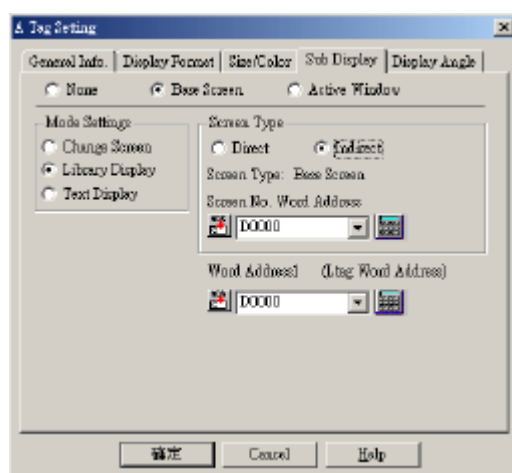
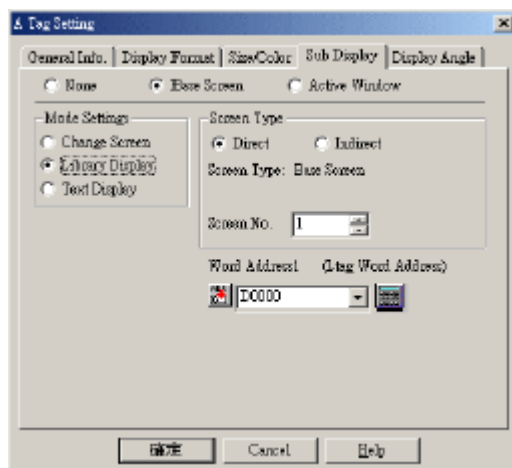
Word Address 1 (L-tag 地址)定义配合的 **L-tag** 存放 **Base Screen** 画面号码的 **Word Address**.

Screen Type – 画面型式

Indirect – 间接,间接指定在画面上显示警报说明画面的 **Base Screen** 号码. 本功能仅提供给 **GP377**, **GP77R** 及 **GP2000** 系列機種.

Screen No. Word Address – 储存画面号码的 **Word** 地址设定储存起始文字画面号码的 **Word** 地址. **Word** 地址的值, 必须是二进制格式.

Word Address 1 (L-tag 地址)定义配合的 **L-tag** 存放 **Base Screen** 画面号码的 **Word Address**.



Text Display – 文字显示

Screen Type – 画面型式

Direct – 直接, 直接指定在画面上显示警报说明画面的 **Text Screen** 号码.

Screen No. - 画面的号码, 输入显示警报说明画面(**Text Screen**)的起始号码..

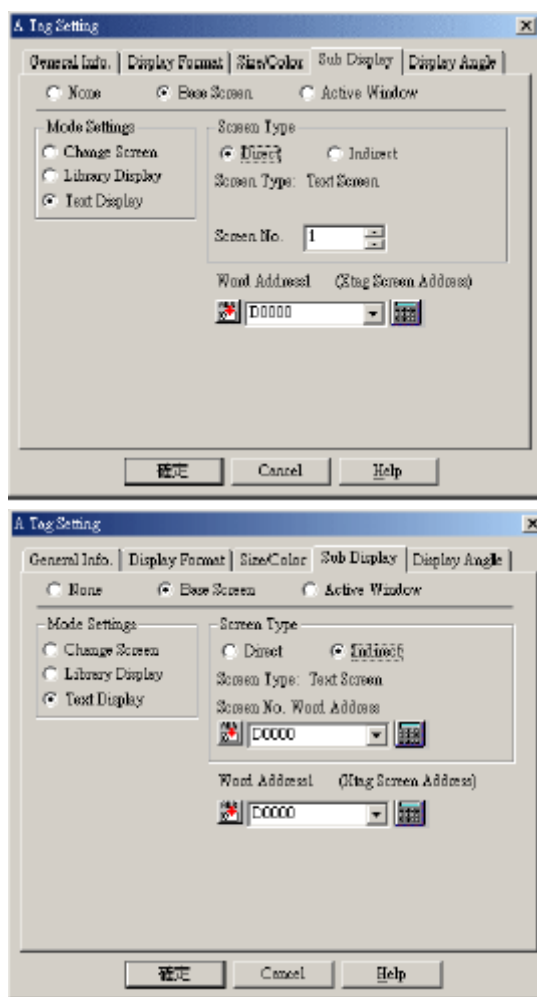
Word Address 1 (X-tag 地址)定义配合的 **X-tag** 存放 **Text Screen** 画面号码 **Word** 地址.

Screen Type – 画面型式

Indirect – 间接, 间接指定在画面上显示警报说明画面的 **Text Screen** 号码. 本功能仅提供给 **GP377**, **GP77R** 及 **GP2000** 系列机种.

Screen No. Word Address – 储存画面号码的 **Word** 地址, 设定储存起始文字画面号码的 **Word** 地址. **Word** 地址的值, 必须是二进制格式.

Word Address 1 (X-tag 地址)定义配合的 **X-tag** 存放 **Text Screen** 画面号码的 **Word** 地址.



◆ Sub Display – Active Window

Change Screen – 改变画面

Screen Type – 画面型式

Direct – 直接，直接指定在画面上显示警报说明窗口的窗口登录号码。

Window Registration No. - 窗口登录号码，输入显示警报说明窗口的登录号码。

Screen Type – 画面型式

Indirect – 间接，间接指定在画面上显示警报说明窗口的登录号码.. 本功能仅提供给 **GP377, GP77R** 及 **GP2000** 系列机种。

Window Registration No. Word Address – 储存窗口登录号码的 **Word** 地址设定
储存起始窗口登录号码的 **Word** 地址. **Word** 地址的值，必须是二进制格式。

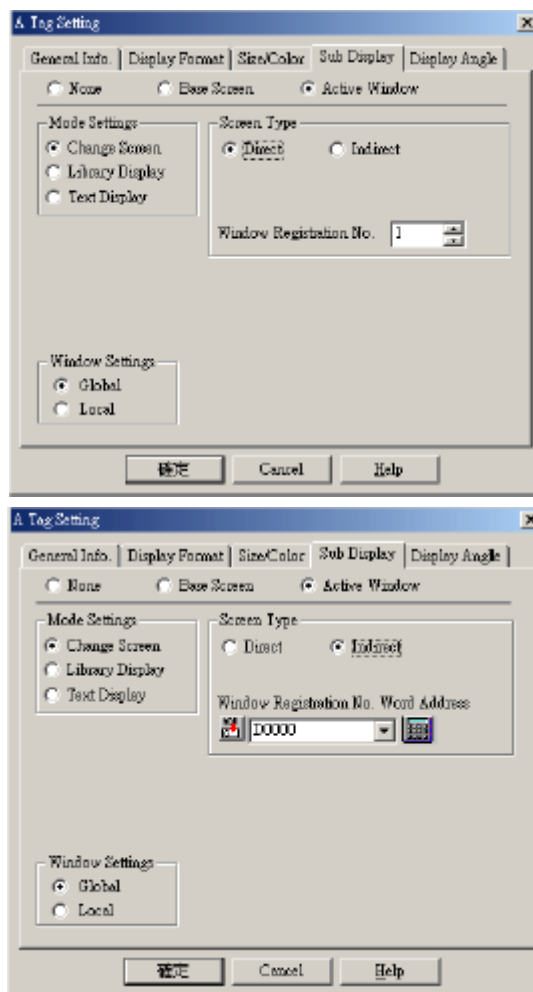
Window settings – 窗口设定。

Global – 共享，共享显示窗口。

Window settings – 窗口设定。

Local – 非共享的，非共享显示窗口。

Word Address2(U-tag Word Address) – **U-tag** 地址，定义配合的 **U-tag** 存放窗口画面号码的 **Word** 地址。



Library Display

Screen Type – 画面型式

Direct – 直接，直接指定在画面上显示警报说明画面的 **Base Screen** 号码。

Screen No. - 画面的号码，输入显示警报说明画面(**Base Screen**)的起始号码。

Word Address 1 (L-tag 地址)定义配合的 **L-tag** 存放 **Base Screen** 画面号码的 **Word Address**。

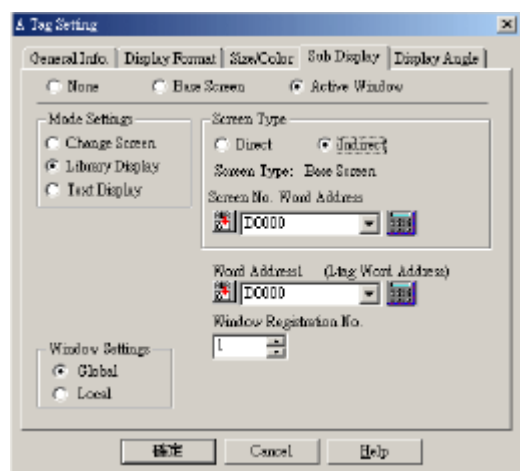
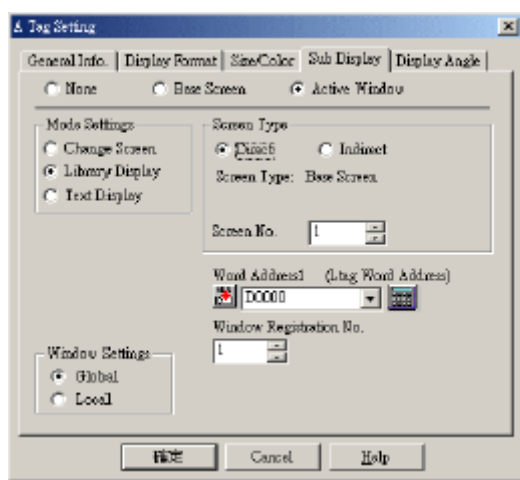
Window Registration No. - 窗口登录号码，输入警报说明窗口的登录号码。

Indirect – 间接，间接指定在画面上显示警报说明画面的 **Base Screen** 号码。本功能仅提供给 **GP377**, **GP77R** 及 **GP2000** 系列机种。

Screen No. Word Address – 储存画面号码的 **Word** 地址，设定储存起始文字画面号码的 **Word** 地址。**Word** 地址的值，必须是二进制格式。

Word Address 1 (L-tag 地址)定义配合的 **L-tag** 存放 **BaseScreen** 画面号码的 **Word Address**。

Window Registration No. - 窗口登录号码，输入警报说明窗口的登录号码



Text Display – 文字显示

Screen Type – 画面型式

Direct – 直接，直接指定在画面上显示警报说明画面的 **Text Screen** 号码。

Screen No. - 画面的号码，输入显示警报说明画面(**TextScreen**)的起始号码。.

Word Address 1 (X-tag 画面地址)定义配合的 **X-tag** 存放 **Text Screen** 画面号码的 **Word** 地址。

Window Registration No. - 窗口登录号码，输入警报说明窗口的登录号码。

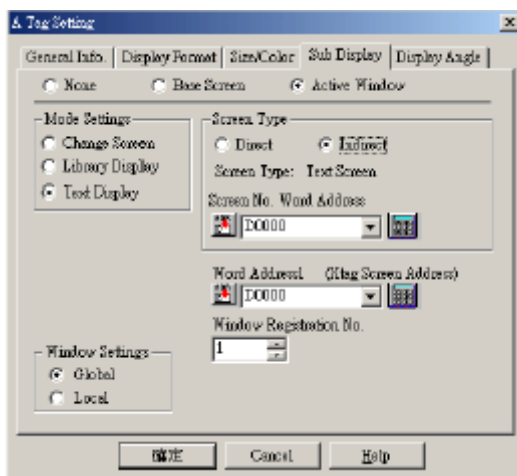
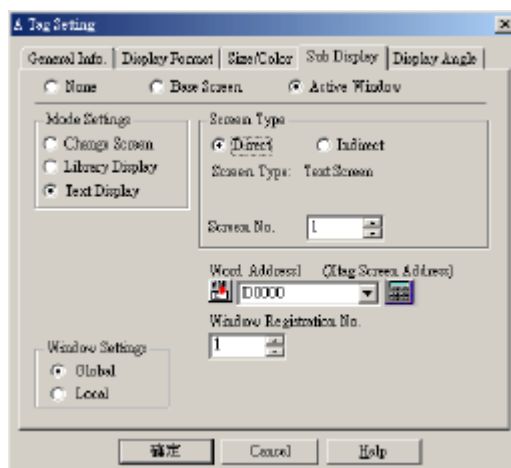
Screen Type – 画面型式

Indirect – 间接，间接指定在画面上显示警报说明画面的 **Base Screen** 号码。本功能仅提供给 **GP377**, **GP77R** 及 **GP2000** 系列机种。

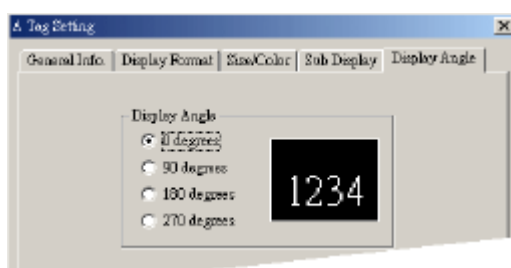
Screen No. Word Address – 储存画面号码的 **Word** 地址，设定储存起始文字画面号码的 **Word** 地址。 **Word** 地址的值，必须是二进制格式。

Word Address (X-tag 画面地址)定义配合的 **X-tag** 存放文字画面号码的 **Word Address**。

Window Registration No. - 窗口登录号码，输入警报说明窗口的登录号码。



- **Display Angle** – 显示角度，设定显示角度为 0, 90, 180 或 270 度。



当设定与 A-tag 配合使用的 U-tag, L-tag 或 X-tag, 请将 U-tag 及 L-tag 指定 Indirect(间接)窗口区及 X-tag 的 Mode/Word/Display Start Line 为 Indirect(间接), 并确定使用的资料格式为 Bin(二进制).

以 Text Screen(文字画面)或 Base Screen(基本画面)作为警报说明画面, 请不要再作其它用途.

11.2 Q-TAG 警报摘要显示

11.2.1 关于 Q-tag

- 使用以下的程序, 来建立 **Q-tag** 警报摘要显示
- 在 **Alarm Editor** 登录警报摘要.
- 使用 **GP** 系统设定中的 **Q-tag** 项来设定 **Q-tag**
- 在 **Base Screen** 上建立 **Q-tag**.
- 在 **Alarm Editor** 登录所有的警报讯息摘要
- **Q-tag** 的日期(**Date**), 触发时间(**Trigger**), 警报摘要讯息(**Alarm Summary**), 确认(**Ack**)及回复(**Recover**)时间的字段距离可在 **GP** 的系统设定中设定, 换句话说, 所有的画面的字段距离可使用相同的设定, 当然也可个别设定.
- 控制警报(**Alarm**)的触发方式, 可使用 **Bit** 或 **Word** 资料.
- 警报摘要讯息, 可以 **Block**(区块)方式来控制, 例如, 一组使用作严重的警报摘要讯息, 另一组使用作监控的警报摘要讯息, 而以 **Block**(区块)方式显示. 最多可设置 3 个 **Block**(区块). 使用 **Alarm Editor** 来指定显示的 **Block**(区块). 而这个功能仅提供给 **GP377**, **GP77R** 及 **GP2000** 机种.
- 当使用 **Alarm Editor** 输入警报摘要讯息及控制触发的外围地址, 外围地址可指定是 **Bit** 或 **Word** 地址, 亦可同时使用.
- 可设定 **Bit** 触发方式是由 **0 1** 或 **1 0**, 当 **Bit** 变为原始状态时, 则为回复时间.
- **Word** 的触发方式为 **Word Address** 的值等于 **Alarm Editor** 设定的值时, 便触发警报, 当不等于时, 为回复时间.
- 可显示的警报摘要讯息个数为 **768**, 依据设定的警报摘要讯息个数储存在 **SRAM** 里, 当显示的警报摘要讯息个数超出范围, 超出范围的警报摘要讯息将不显示, 保留原有的旧资料.
- 可登录在 **Alarm Editor** 内的警报摘要讯息个数, 因 **GP** 的机种而有所不同

GP 机种	可登录的警报摘要讯息个数
GP270, GPH70, GP370	512
非上列的 GP70 机种, GP37W2 GP377, GP77R 系列机种	768
GP2000 系列机种	2048

- **Number**(组号)可在 **Alarm Editor** 设定, 最大的 **Group Number** 就是最大的登录讯息数, 对 **GP2000** 系列机种, 最大的 **Group Number** 为 **2012**, (这是在 **LS** 区内的最大使用者区), 当所以设定了最大的警报讯息数, **Group Number**(组号)无法对每个警报讯息作设定. 请确认检查 **LS** 区的大小.
- 每个画面可显示的行数与每行的字数与 **GP** 的机种及显示的字号有关.
- 触发, 清除及确认的显示大小, 颜色属性可以设定.
- 可选择日期及时间的显示格式.
- 配合 **T-tag** 使用, 可建立 **Start**(开始), **Up**(上一行), **Down**(下一行), **Ack**(确认), **Ack All**(确认所有), **Delete**(删除), **Delete All**(删除所有), **End**(结束), **Sub-Screen**(警报说明的子画面)

- 当未在 **Alarm Editor** 中使用 **Block** 功能警报讯息触发, 将依照 **Q-tag** 设定模式 (**Active, History, Log**)来显示
- 当在 **Alarm Editor** 中使用 **Block** 功能警报讯息触发, 将依照 **Q-tag** 设定模式 (**Block 1, Block 2, Block 3**)来显示. 警报讯息将显示在设定相同 **Block** 的 **tag** 内
- **Q-tag** 的日期, 触发时间, 警报摘要讯息, 确认及回复时间的字段距离可在 **GP PRO/PBIII** 软件的 **GP System Settings** 中设定
- 依照选择的警报讯息, 可显示如何修复警报的说明子画面(**Sub-Screen**).
- 说明子画面(**Sub-Screen**), 可使用 **L-tag**(**Library** 显示), **X-tag**(文字画面显示) 或窗口显示

11.2.2 Q-tag 的使用限制

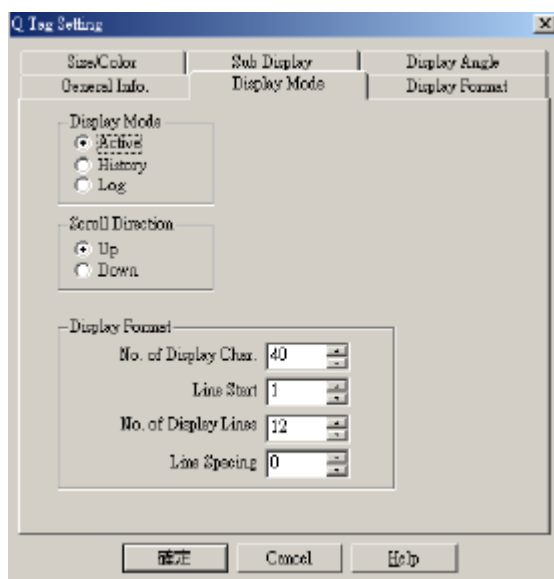
- ◆ 如果显示的警报讯息超出显示字段, 讯息将不显示.
- ◆ 为了配合 **Q-tag** 所使用的 **T-tag** 功能键, 所以一个画面只能使用一个 **Q-tag**.
- ◆ 使用 **T-tag** 设定的功能键, 必须和 **Q-tag** 是在同一个画面

11.2.3 使用 Q-Tag

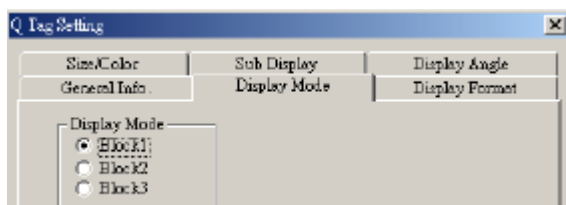
- **General Info** – 一般信息, 提供 **Q-tag** 目前设定的相关信息.
Tag Name - 卷标名称, 须小于等于 **5** 个英文字符或 **2** 个汉字字符.
Description - 叙述, 须小于等于 **20** 个英文字符或 **10** 个汉字字符.



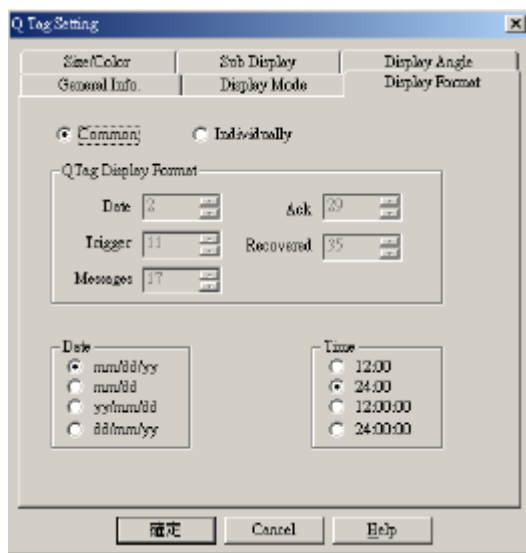
- **Display Mode** – 显示模式, 选择显示格式为: **Active, History** 或 **Log**. 如果在 **Alarm Editor** 选择了 **Block**(区块)功能, 则选择显示区块为 **Block1, 2** 或 **3**. 而这个功能仅提供给 **GP377, GP77R** 及 **GP2000** 机种.
Scroll Direction – 屏幕卷动方向, **Up Down** 选择屏幕卷动方向是向上或向下.
- **Display Mode**
No of Display Char. : 显示字符个数, 指定每一个 **Alarm Message** 的最大显示字符串长度. 单位为半角字元.(一个中文字, 占二个字符)
Line Start : 起始行的显示位置, 指定起始行的显示位置.
No of Display Line : 显示的行数, 指定一个画面的显示行数.
Line Spacing : 行距, 每一行之间的行距. 设定范围为 **0-7**, 单位为 **Dot**(点).



当在 Alarm Editor 中选择了 Block 模式时显示.



■ Display Format – 显示格式



Common – 共享，使用这个选项，显示的格式在*GP System Settings* 的 Q-tag 设定，换句话说，使用这个选项，在这个 **Project** 里，**Q-tag** 显示格式与*GP System Settings*一样。
Individually – 个别设定，选择这个选项，显示格式则在 **Q-tag** 里，个别设定。

Q-tag Display Format – 显示格式

Date 日期

Trigger 触发时间

Message 警报讯息

Ack 确认时间

Recovered 回复时间

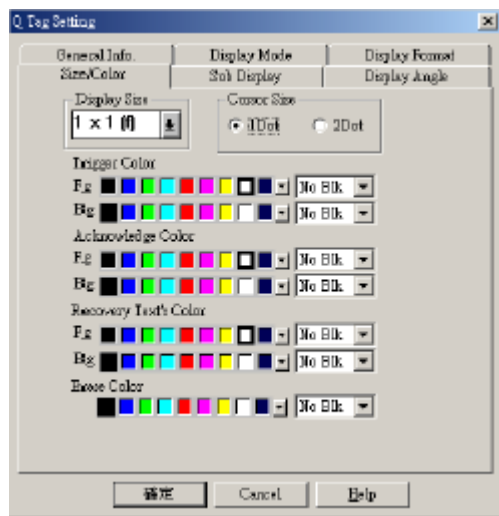
位置距离从最左边开始计算，以 8 个点(一个字符)为单位。当设定位置的字段里被设定为 0

时，则这一个字段将不显示.GP-270 不支持此一功能。

Date - 日期，指定日期的显示格式。

Trigger - 触发时间，指定触发时间的显示格式。

■ **Size/Color** – 大小颜色



Display Size – 显示字号，显示字号

Cursor Size – 光标大小, 指定光标线的宽度.

Trigger Color – 触发时间显示颜色，触发时间的显示颜色，可指定 **Fg** 前景，**Bg** 背景及 **Blk** 闪烁属性。

Acknowledge Color – 确认显示颜色，确认时间的显示颜色，可指定 **Fg** 前景，**Bg** 背景及 **Blk** 闪烁属性。

Recovered Text's Color – 回复颜色，回复时的文字显示颜色，可指定 **Fg** 前景，**Bg** 背景及 **Blk** 闪烁属性。

Erase Color – 清除颜色，清除警报信息的显示颜色，单色机种请选择黑色。

■ **Sub Display** – 警报说明画面

None – 不使用, 使用这个选项, 不使用警报说明子窗口画面.

Base Screen – 基本画面, 这个选项, 使用基本画面来作警报说明画面. 本功能亦可使用作画面的改变显示.

当以 **X-tag** 作为警报文字说明画面或以 **L-tag** 作为警报说明窗口画面，在输入控制的 **Word Address** 时，仅能使用 **LS Address**。

Active Window – □□□□□□,□□□□□□□□□□□□□□□□□□.

GP-270 不支持警报说明画面功能.

■ **Sub Display/Base Screen - 警报说明画面/基本画面**



Mode Settings – 模式设定

Change Screen – 改变画面, 改变显示画面到另一个画面, 如同一般换页动作. 改变画面的号码在 **Alarm Editor** 里设定.

Library Display – Library 显示, 以 **L-tag** 设定警报说明画面显示位置.

Text Display – 文字显示, 以 **X-tag** 设定警报文字说明画面显示位置.

Offset – 偏差值, 定义一个加到 **Word Address** 的偏差值. 如不使用偏差值, 则设此字段的值为 **0**.

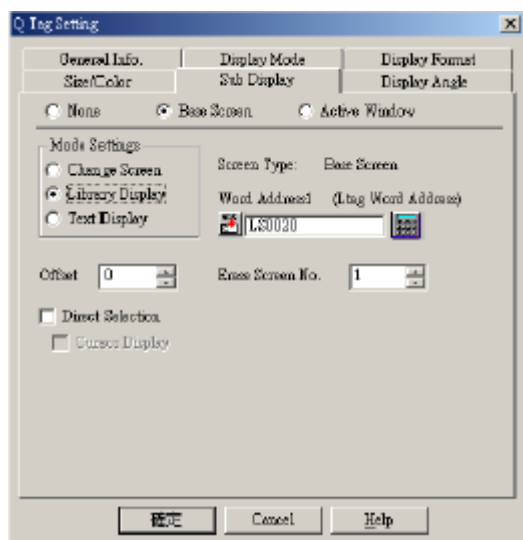
Direct Selection – 直接选择, 用这个选项,

可按下屏幕显示的警报讯息，并显示警报说明画面

Cursor Display – 光标显示, 这个选项将在选择的警报讯息上出现光标, 当光标出现时,

你可检视警报讯息的内容.

使用 **Library** 显示警报画面, 必须在 **Q-tag** 所在的画面上放置一个 **L-tag**. 而 **L-tag** 的必须指定画面方式为 ***Indirect***(间接), **Data Format**(资料格式)为 **Bin**(二进制). 使用 **Text**(文字)显示警报画面, 必须在 **Q-tag** 所在的画面上放置一个 **X-tag**. 而 **X-tag** 的必须指定画面方式为 ***Indirect***(间接), **Data Format**(资料格式)为 **Bin**(二进制).



Base Screen – 基本画面

Mode Settings – 模式设定

Library Display

Word Address 1 (L-tag 地址),定义 L-tag 的 **Base Screen** 画面号码的 **Word Address**, 使用 **AlarmEditor** 输入警报说明 画面号码.

Erase Screen No. – 清除画面号码, 当在 **Alarm Editor** 里设定的的警报说明画面号码为 **0** 时, 呼叫本警报讯息时, 将清除先前显示的警报说明画面. 所以请制作一个以黑色填满的方形画面(**Base Screen**)作为清除画面.

Base Screen – 基本画面

Mode Settings – 模式设定

Text Display

Word Address 1 (X-tag 地址),定义 X-tag 的 **Text Screen** 画面号码的 **Word Address**, 使用 **Alarm Editor** 输入警报说明画面号码.

Erase Screen No. – 清除画面号码当在 **Alarm Editor** 里设定的的警报说明画面号码为 **0** 时, 呼叫本警报讯息时, 将清除先前显示的警报说明画面. 所以请制作一个以黑色填满的方形画面(**Base Screen**)作为清除画面



■ **Sub Display/Active Window** – 警报说明画面/动态窗口,

Mode Settings – 模式设定

Change Screen – 改变画面, 这个设定开启窗口显示说明画面. 窗口显示说明画面的号码在 **Alarm Editor** 内输入.

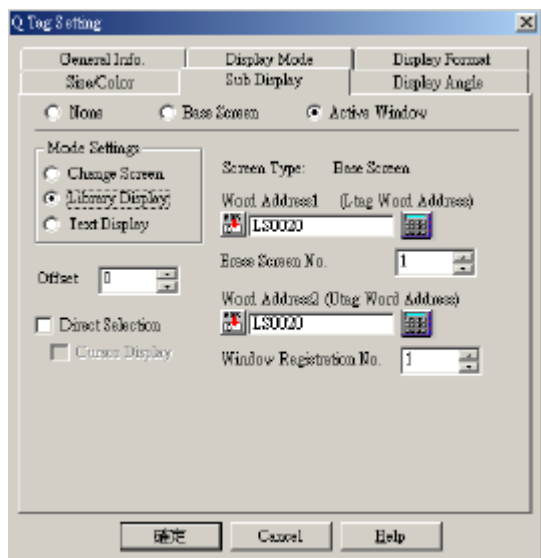
Library Display – **Library** 显示以 **L-tag** 设定警报说明画面显示位置.

Text Display – 文字显示, 以 **X-tag** 设定警报文字画面显示位置.



Word Address 2 (U-tag 地址)定义显示窗口画面号码的地址. 当使用 **Local Windows** 控制, 则使用 **U-tag** 定义窗口号码地址, 若使用 **Global Window** 则窗口画面的号码地址为 **LS0016**.

- 使用 **Active Window** 显示警报画面, 必须在 **Q-tag** 所在的画面上放置一个 **U-tag**. 而 **U-tag** 的必须指定画面方式为 ***Indirect***(间接), **Data Format**(资料格式)为 **Bin**(二进制).
- 使用 **Library** 显示警报画面, 必须在 **Window Registration** 设置一个 **L-tag**. 而 **L-tag** 的必须指定画面方式为 ***Indirect***(间接), **Data Format**(资料格式)为 **Bin**(二进制).
- 使用 **Text**(文字)显示警报画面, 必须在 **Window Registration** 设置一个 **X-tag**. 而 **X-tag** 的必须指定画面方式为 ***Word***, **Data Format**(资料格式)为 **Bin**(二进制).



Sub Screen 画面显示.

Erase Screen No. – 清除画面号码

当在 **Alarm Editor** 里设定的的警报说明画面号码为 **0** 时, 呼叫本警报讯息时, 将清除先前显示的警报说明画面. 所以请制作一个以黑色填满的方形画面(**Base Screen**)作为清除画面.

Window Registration No. – 窗口登录号码, 输入要显示的窗口的登录号码.

当在 **Alarm Editor** 中输入其中一个警报, 设定了 **Sub Screen No** 及 **Window**

Registration No. 为 **0**, 为了避免警报说明画面显示第 **0** 画面, 请指定 **Erase**

Screen No. – 清除画面号码到 **L-tag** 的地址里

Active Window – 动态窗口

Mode Settings – 模式设定

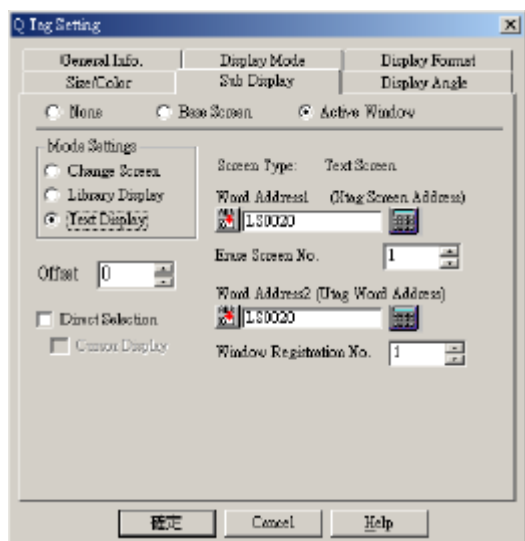
Library Display – **Library** 显示

Word Address 1 (L-tag 地址)定义 **L-tag** 的 **Base Screen** 画面号码的 **Word Address**, 使用 **Alarm Editor** 输入警报说明画面号码.当在 **Alarm Editor** 中输入其中一个警报, 设定了 **Sub Screen No** 及 **Window**

Registration No. 为 **0**, 为了避免警报说明画面显示第 **0** 画面, 请指定 **Erase Screen No.** – 清除画面号码到 **L-tag** 的地址里.

当警报说明画面(**Sub Screen**)设定为 **Active Window**(动态窗口)时, 设定 **Window**

Registration No.(窗口登录号码)为 **0**, 以预防目前的 **Base Screen** 切换到其它画面时, 从



Window(动态窗口)时, 设定

Window Registration No.(窗口登录号码)为 **0**, 以预防目前的 **Base Screen** 切

换到其它画面时, 从 **Sub Screen** 画面显示

Active Window – 动态窗口

Mode Settings – 模式设定

Text Display – 文本显示

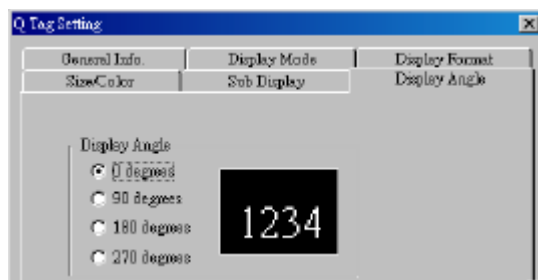
Word Address 1(X-tag 画面地址)定义 **X-tag** 的文字画面号码的 **Word Address**, 使用 **Alarm Editor** 输入警报说明文字画面号码.

Erase Screen No. – 清除画面号码, 当在 **Alarm Editor** 里未设定的警报说明画面号码时, 呼叫本警报讯息时, 将清除先前显示的警报说明文字画面.

Window Registration No. – 窗口登录号码, 输入要显示的窗口的登录号码.

当警报说明画面(**Sub Screen**)设定为 **Active**

■ **Display Angle** – 显示角度, 设定显示角度.



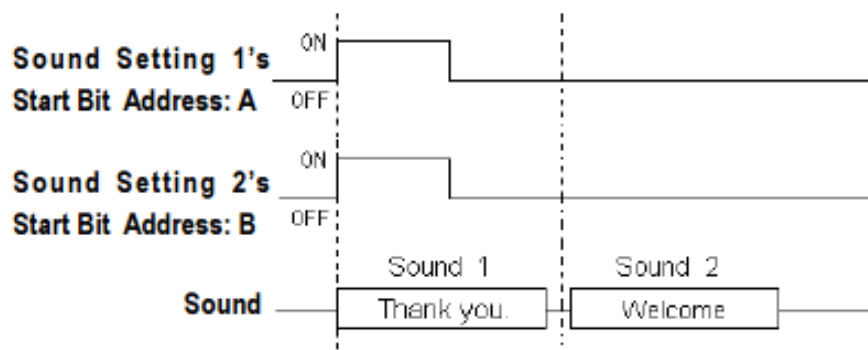
12 声音输出

12.1 概述

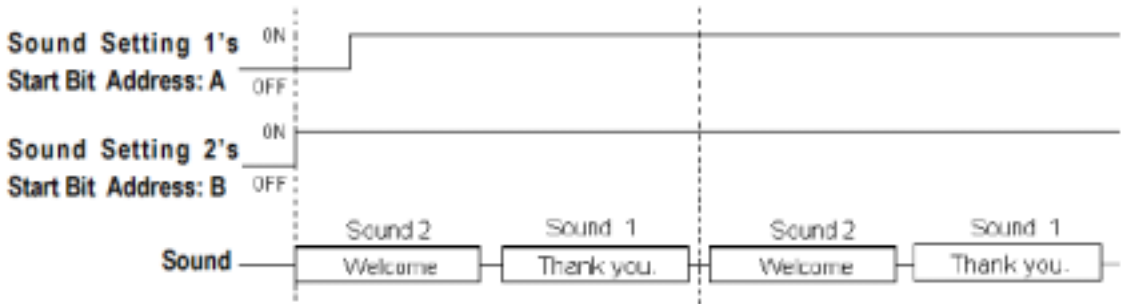
PLC 的一个位为 ON 时，指定的音频数据就从触摸屏的音频输出口输出，通过外部的扬声器系统播放出来。该功能支持 GP77R（GP-377R 除外）、GP2400、GP2500、GP2600 系列，GP-477R、GP-577R、GP2501 系列应用该功能需要外部配置。

- GP 的声音文件是 PCM、11KHZ 的 WAV 数据。
- 要把 WAV 数据转换成 GP 声音数据，请选择工程管理器的[Sound Settings]/[Create Sound Data Selection]选项。
- 声音数据可以存储在 GP 内部存储器或者外部 CF 卡中，如果采用 CF 卡，可以比内部存储器存放更多的声音数据。
- 下面是三种声音输出形式：
 1. 重复播放模式
当触发位置位，声音数据反复播放输出，当多重触发位置位，所有的声音数据反复播放输出。
 2. 播放模式(Bit on)。
当触发位置位，声音数据仅仅播放一次。
 3. 播放模式(Bit off)
当触发位置位，声音数据仅仅播放一次，然后触发位复位。该模式仅仅适用于 LS 区。
- 当触发位复位，声音输出随即取消。

注意：1 当同一时间有超过一个的触发位被触发时，声音数据通常按照声音设置的顺序播放出来，但是根据数据传送时间的不同，顺序可能也不同。



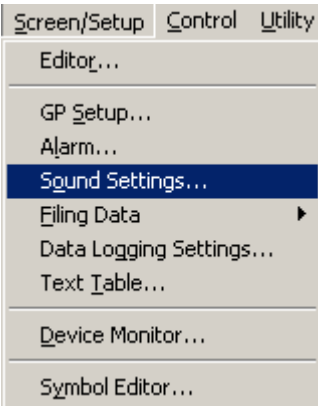
注意：2 在一个声音数据在循环播放过程中，另外一个循环播放模式的声音数据的触发位被触发，声音数据按照声音设置的顺序播放出来。



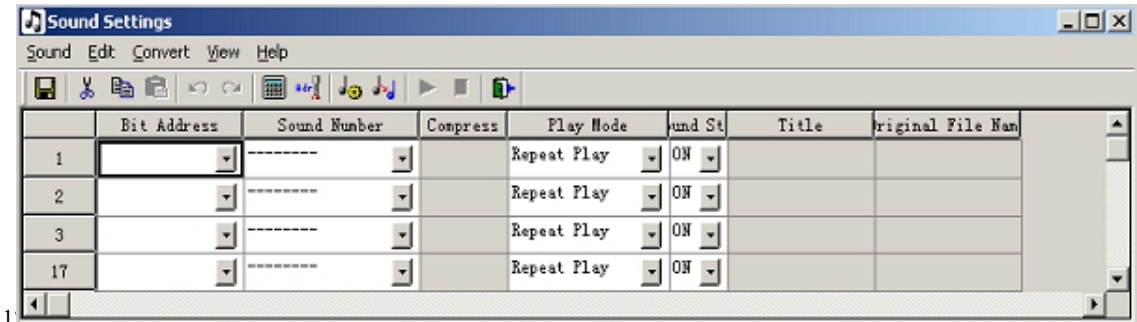
- 例 1：当循环模式设置的声音数据 1、2 和 3 的触发位在同一时间触发，声音输出顺序为：
1→2→3→1→2→3→1→2→3……
- 例 2：当循环模式设置的声音数据 1、2 和 3 的触发位按照 3、2、1 的触发位顺序逐一触发，声音输出顺序为：3→2→1→1→2→3→1→2→3……

12.2 声音输出设置

在工程管理器的[Screen/Setup]下选中[Sound Settings...]项。弹出如下操作界面。



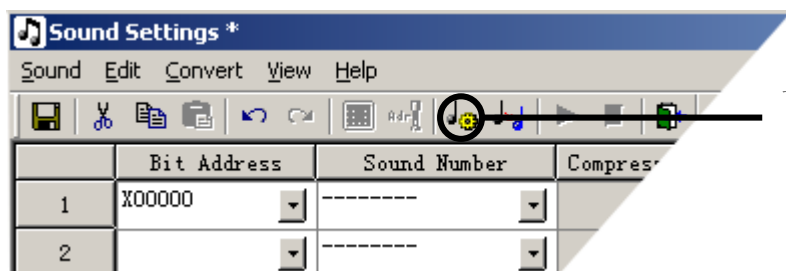
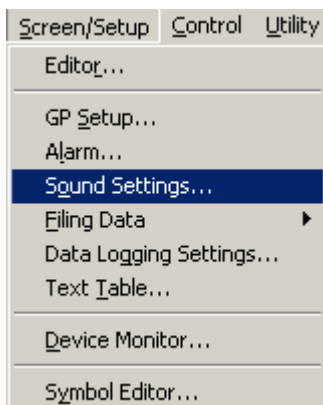
- Bit Address(位地址)
设置输出声音数据的触发位地址。
- Sound Number(音频编号)
给从 WAV 文件转换过来的音频文件编号。
- Compress(压缩)
显示声音数据的压缩状态，压缩或者不压缩。
- Play Mode(播放模式)
Repeat Play:循环播放模式
Play:播放模式
Play(Bit off):播放模式(带复位)
- Sound Stop(音频取消)
选择音频的取消与否，取消或者不可取消。
- Title(标题)
显示音频数据的标题。
- Original File Name(源文件名)
显示转化成 GP 音频文件的源文件名。



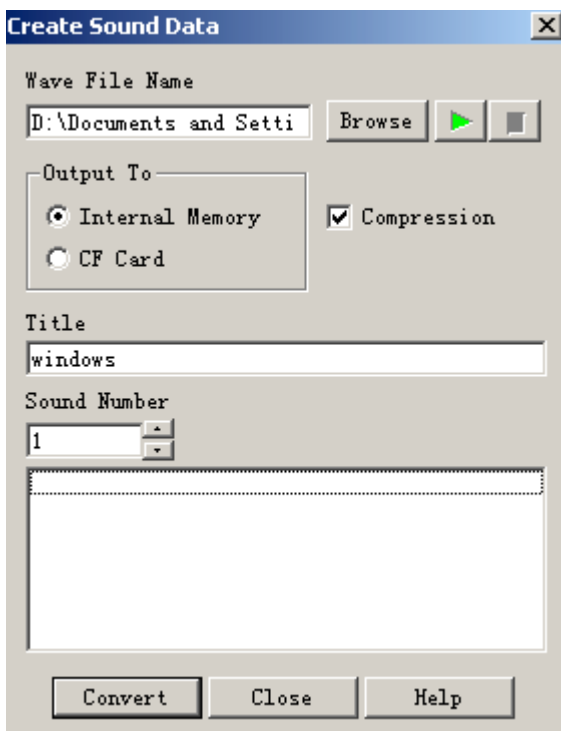
12.3 声音输出设置范例。

12.3.1 创建音频文件。

选择工程管理器[Screen/Setup]下的[Sound Settings...]选项



单击该按钮



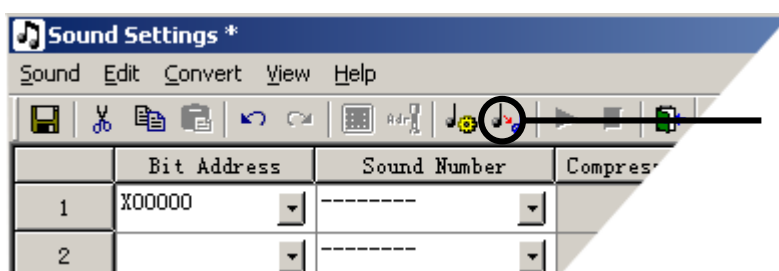
单击“Browse”按钮，然后选择需要转换的 WAV 文件，然后设定标题和编号，最后单击“Convert”按钮。

最后单击“Close”按钮退出转换。

注意:

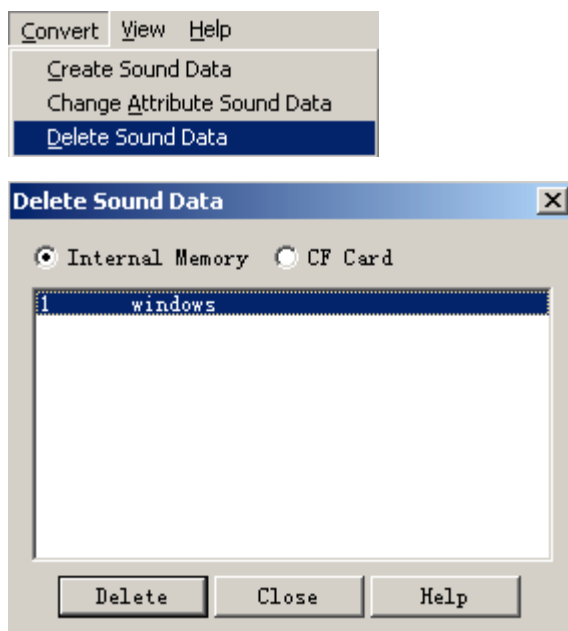
- 1) 标题栏可以输入三十个单字节或者十五个双字节。
- 2) 若需要从 CD 中录入音频数据, 请选择带有 **daf** 扩展名的音频数据
- 3) GP 内部存储器可容纳大约 50k 经过压缩的音频数据 (大概播放约 8 秒钟)。若采用 CF 卡, 则要根据 CF 卡的容量来计算能录入音频的时间。
- 4) GP 转换的音频文件是 PCM、11KHz、16 位的单声道的, 可以利用 Windows 操作系统内带的“录音机”来实现转换。

12.3.2 修改音频文件



单击该按钮可以更改已转换音频文件的一些属性。

12.3.3 删除音频文件



选中在[Convert]菜单下[Delete Sound Data]选项, 可以对现有的音频文件实现删除操作。