

### § 5.4 编辑并调用无参功能（FC）

#### ——分部程序设计

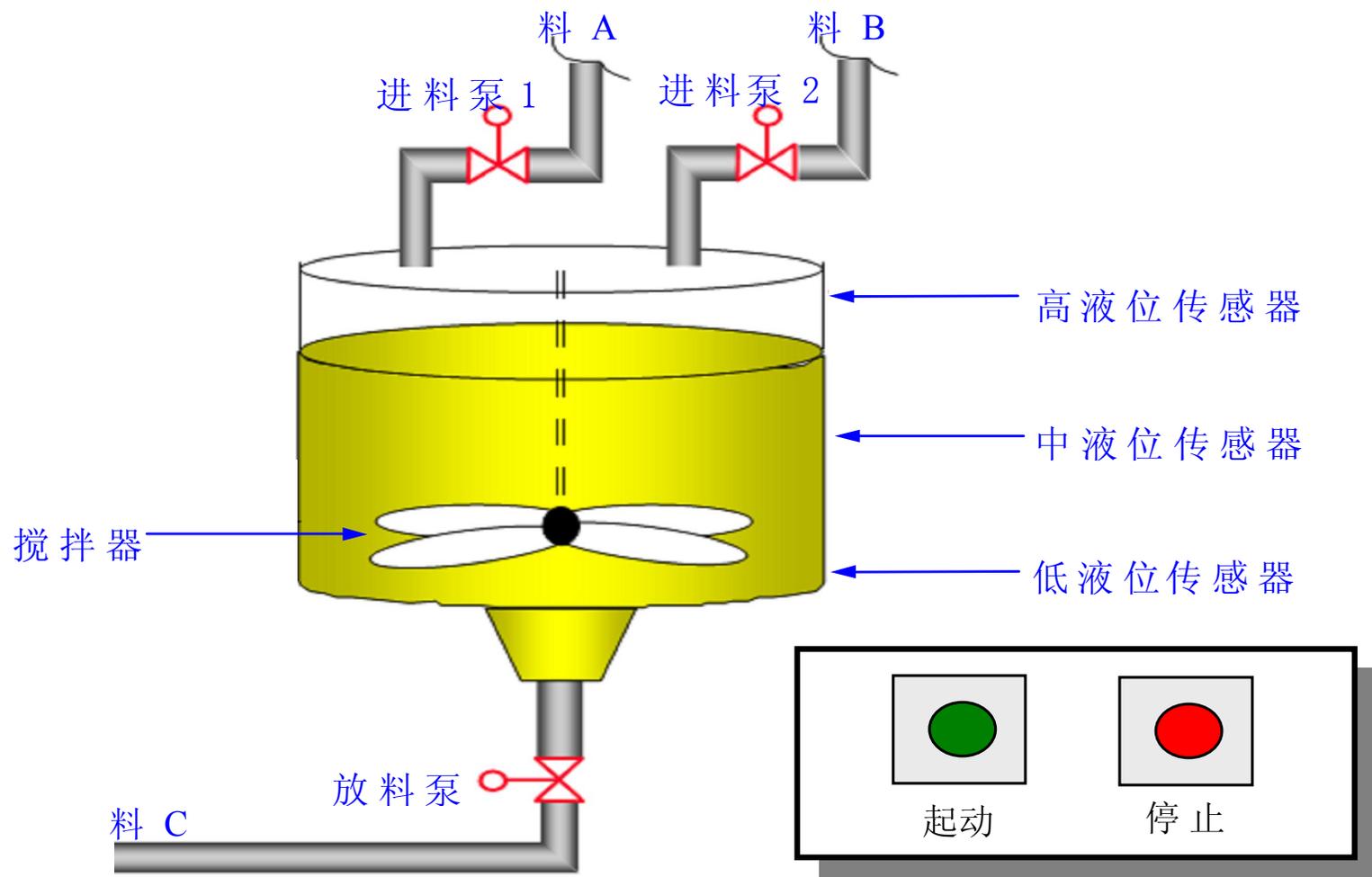
所谓**无参功能**（FC），是指在编辑功能（FC）时，在局部变量声明表不进行形式参数的定义，在功能（FC）中直接使用绝对地址完成控制程序的编程。这种方式一般应用于分部式结构的程序编写，每个功能（FC）实现整个控制任务的一部分，不重复调用。

#### § 5.4.1 编辑无参功能（FC）

#### § 5.4.2 在OB1中调用无参功能（FC）

## CH5 S7程序结构与程序设计

### 【例5-4-1】 搅拌控制系统程序设计——使用开关量。



返回本节

## CH5 S7程序结构与程序设计

### ►控制说明

如图所示为一搅拌控制系统，由3个开关量液位传感器，分别检测液位的高、中和低。现要求对A、B两种液体原料按等比例混合，请编写控制程序。

要求：按起动按钮后系统自动运行，首先打开进料泵1，开始加入液料A→中液位传感器动作后，则关闭进料泵1，打开进料泵2，开始加入液料B→高液位传感器动作后，关闭进料泵2，起动搅拌器→搅拌10s后，关闭搅拌器，开启放料泵→当低液位传感器动作后，延时5s后关闭放料泵。按停止按钮，系统应立即停止运行。

## CH5 S7程序结构与程序设计

### § 5.4.1 编辑无参功能（FC）

#### 1. 创建S7项目

按照第3章所介绍的方法，创建S7项目，并命名为“无参FC”，项目包含组织块OB1和OB100。

#### 2. 硬件配置

在“无参FC”项目内打开“SIMATIC 300 Station”文件夹，打开硬件配置窗口，并完成硬件配置。

Slot	Module	Order number	Fi...	MPI address	I address	Q address	Comment
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU315 (1)	6ES7 315-1AF03-0AB0		2			
3							
4	DI32xDC24V	6ES7 321-1BL80-0AA0			0...3		
5	DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0				4...7	
6							

返回本节

## CH5 S7程序结构与程序设计

### 3. 编辑符号表

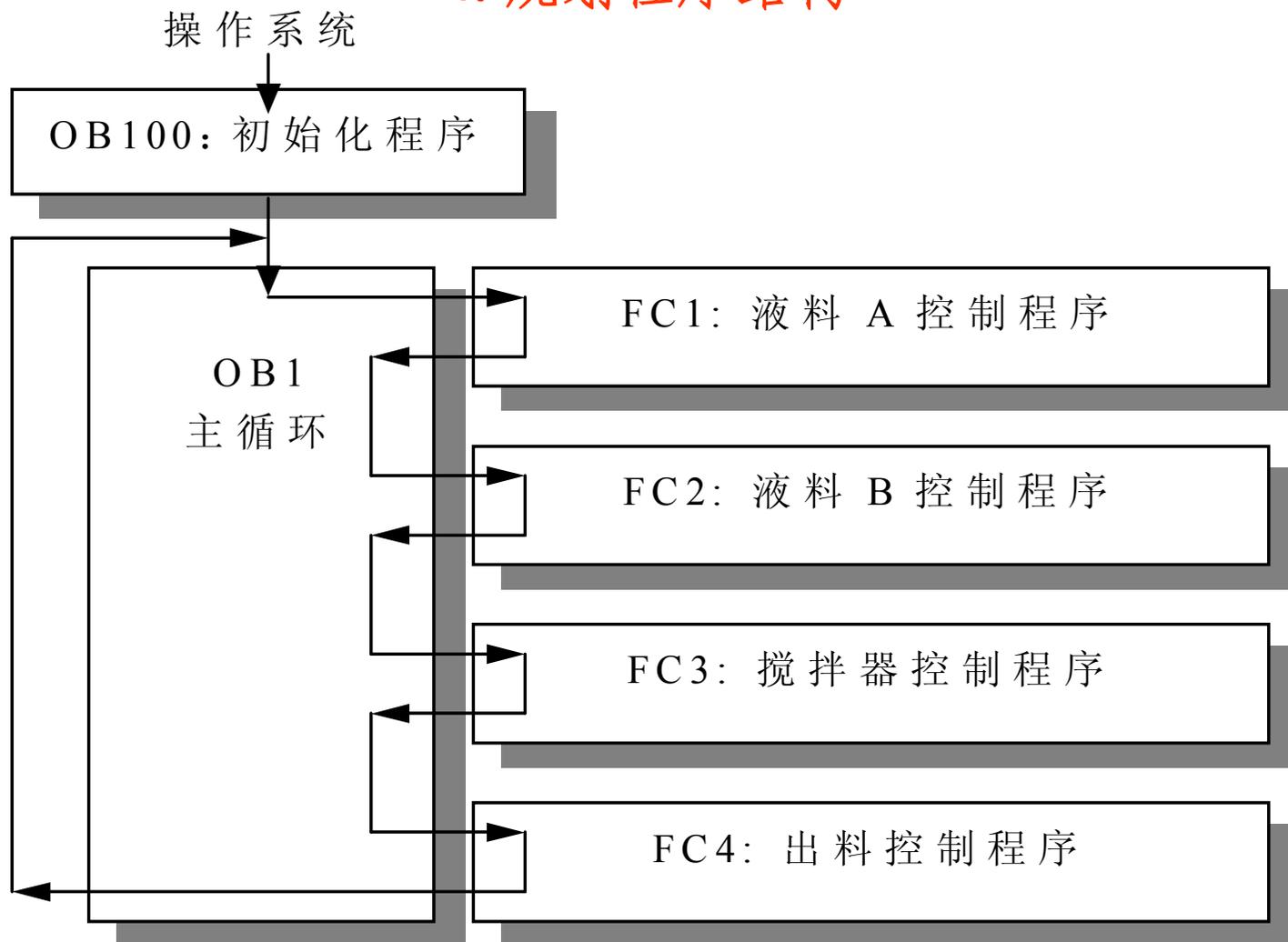


	Status	Symbol	Address	Data type	Comment
4		中液位检测	I 0.3	BOOL	有液料时为"1"
5		低液位检测	I 0.4	BOOL	有液料时为"1"
6		原始标志	M 0.0	BOOL	表示进料泵、放料泵及搅拌器均处于停机状态。
7		最低液位标志	M 0.1	BOOL	表示液料即将放空
8		Cycle Execution	OB 1	OB 1	线性结构的搅拌器控制程序
9		BS	PW 256	WORD	液位传感器-变送器，送出模拟量液位信号
10		DISP	PQW 256	WORD	液位指针式显示器，接收模拟量液位信号
11		进料泵1	Q 4.0	BOOL	"1"有效
12		进料泵2	Q 4.1	BOOL	"1"有效
13		搅拌器M	Q 4.2	BOOL	"1"有效
14		放料泵	Q 4.3	BOOL	"1"有效
15		搅拌定时器	T 1	TIMER	SD定时器，搅拌10s
16		排空定时器	T 2	TIMER	SD定时器，延时5s
17		液料A控制	FC 1	FC 1	液料A进料控制
18		液料B控制	FC 2	FC 2	液料B进料控制
19		搅拌器控制	FC 3	FC 3	搅拌器控制
20		出料控制	FC 4	FC 4	<a href="#">出料泵控制</a>

返回上级

# CH5 S7程序结构与程序设计

## 4. 规划程序结构



返回上级

### 5. 编辑功能（FC）

在“无参FC”项目内选择“Blocks”文件夹，然后反复执行菜单命令【Insert】→【S7 Block】→【Function】，分别创建4个功能（FC）：FC1、FC2、FC3和FC4。由于在符号表内已经为FC1～FC4定义了符号名，因此在创建FC的属性对话框内系统会自动添加符号名。

✓ FC1控制程序

✓ FC2控制程序

✓ FC3控制程序

✓ FC4控制程序

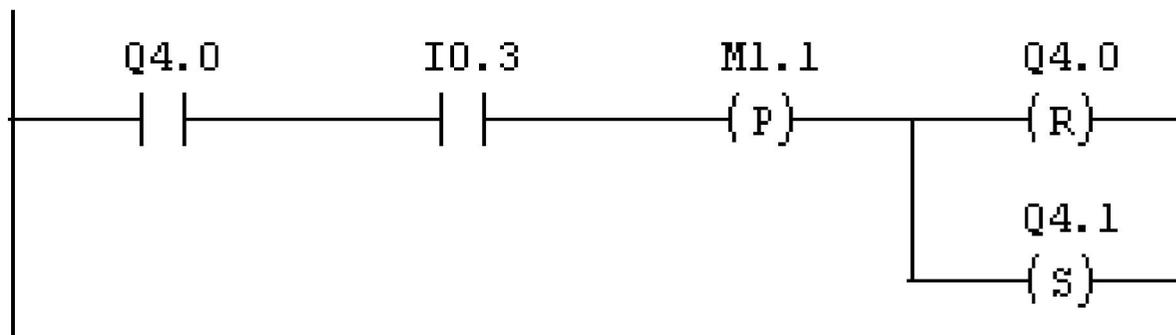
✓ OB100控制程序

## CH5 S7程序结构与程序设计

FC1 : 配料A控制子程序

➤ FC1的控制程序

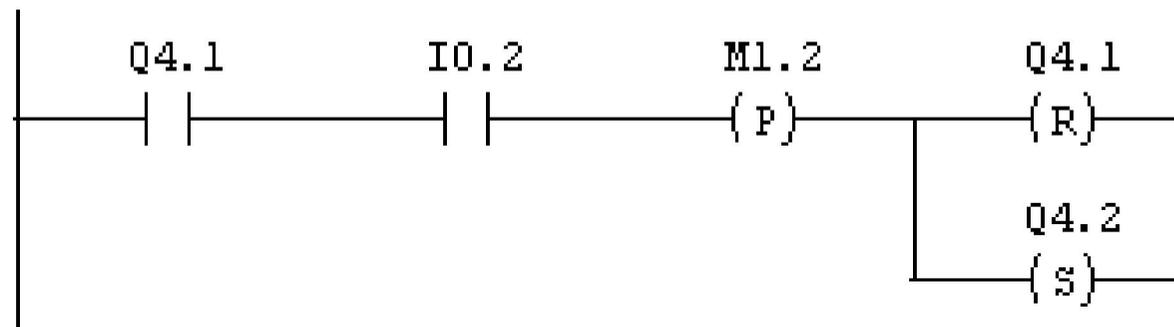
Network 1: 关闭进料泵1, 启动进料泵2



FC2 : 配料B控制程序

➤ FC2的控制程序

Network 1: 关闭进料泵2, 启动搅拌器



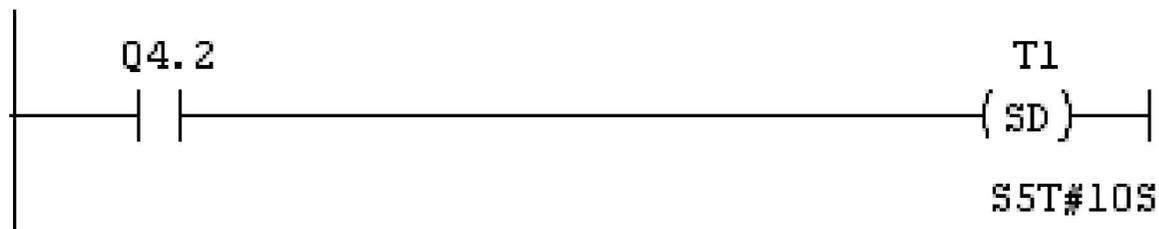
返回上级

# CH5 S7程序结构与程序设计

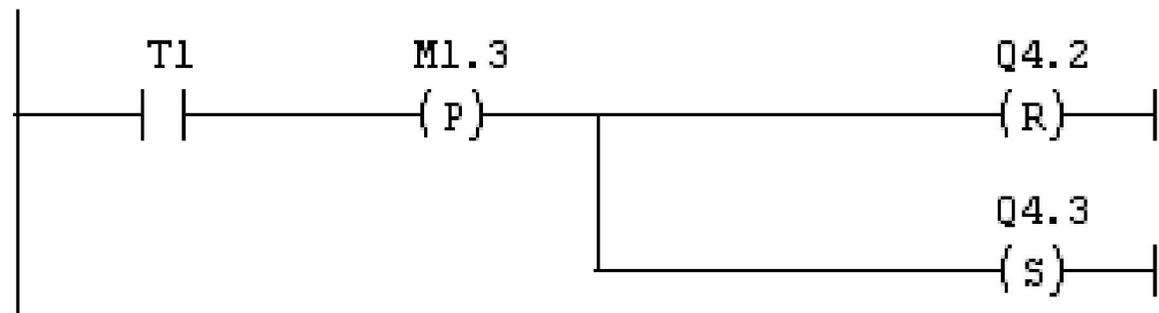
## ► FC3的控制程序

FC3 : 搅拌器控制程序

Network 1: 设置10s搅拌定时



Network 2: 关闭搅拌器, 启动放料泵



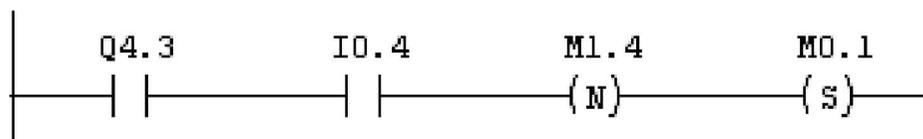
返回上级

# CH5 S7程序结构与程序设计

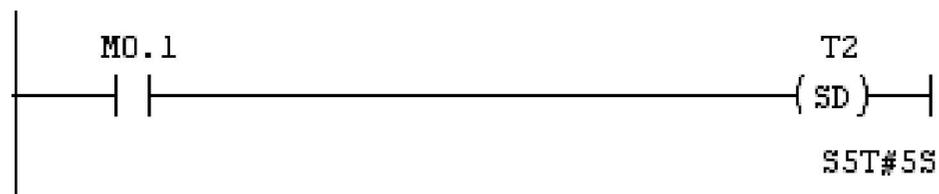
FC4 : 放料控制程序

➤ FC4的控制程序

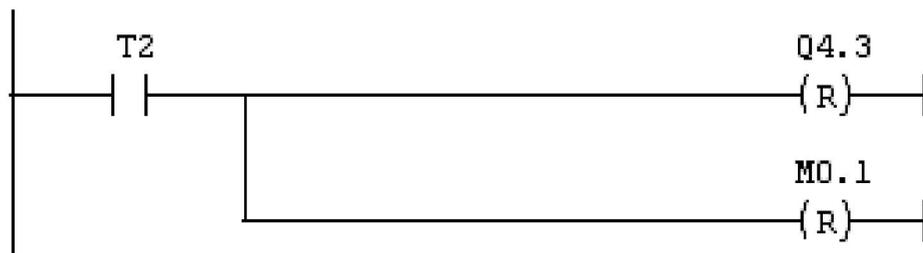
**Network 1:** 设置最低液位标志



**Network 2:** SD定时器, 延时5s



**Network 3:** 清除最低液位标志, 关闭放料泵



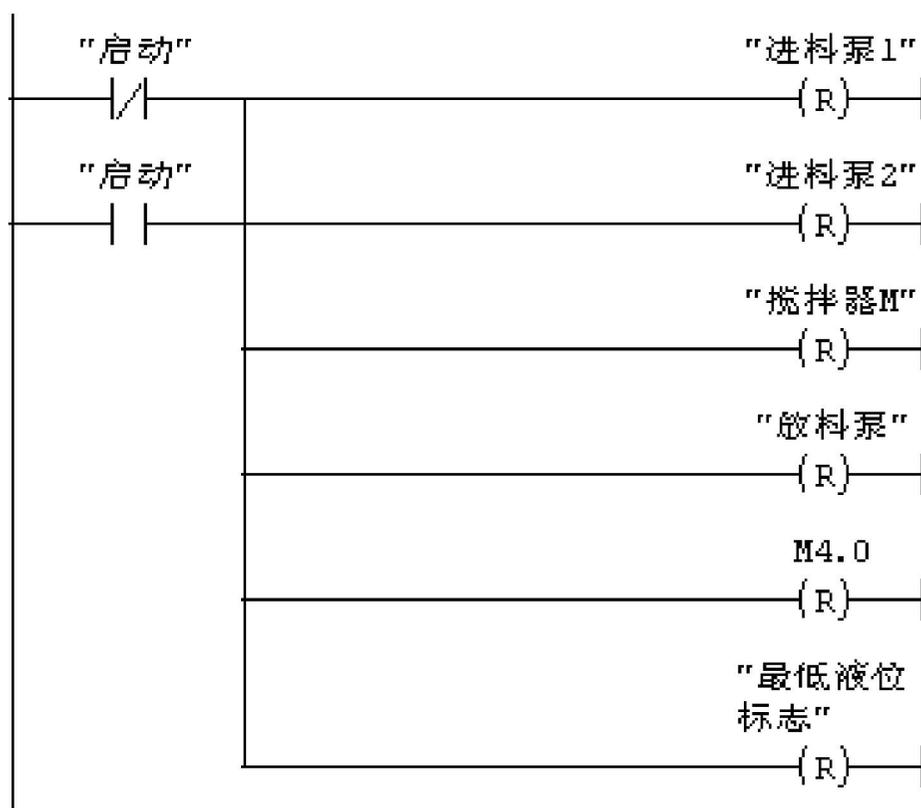
返回上级

## CH5 S7程序结构与程序设计

### ► OB100的控制程序

OB100 : “搅拌控制程序-完全启动复位组织块”

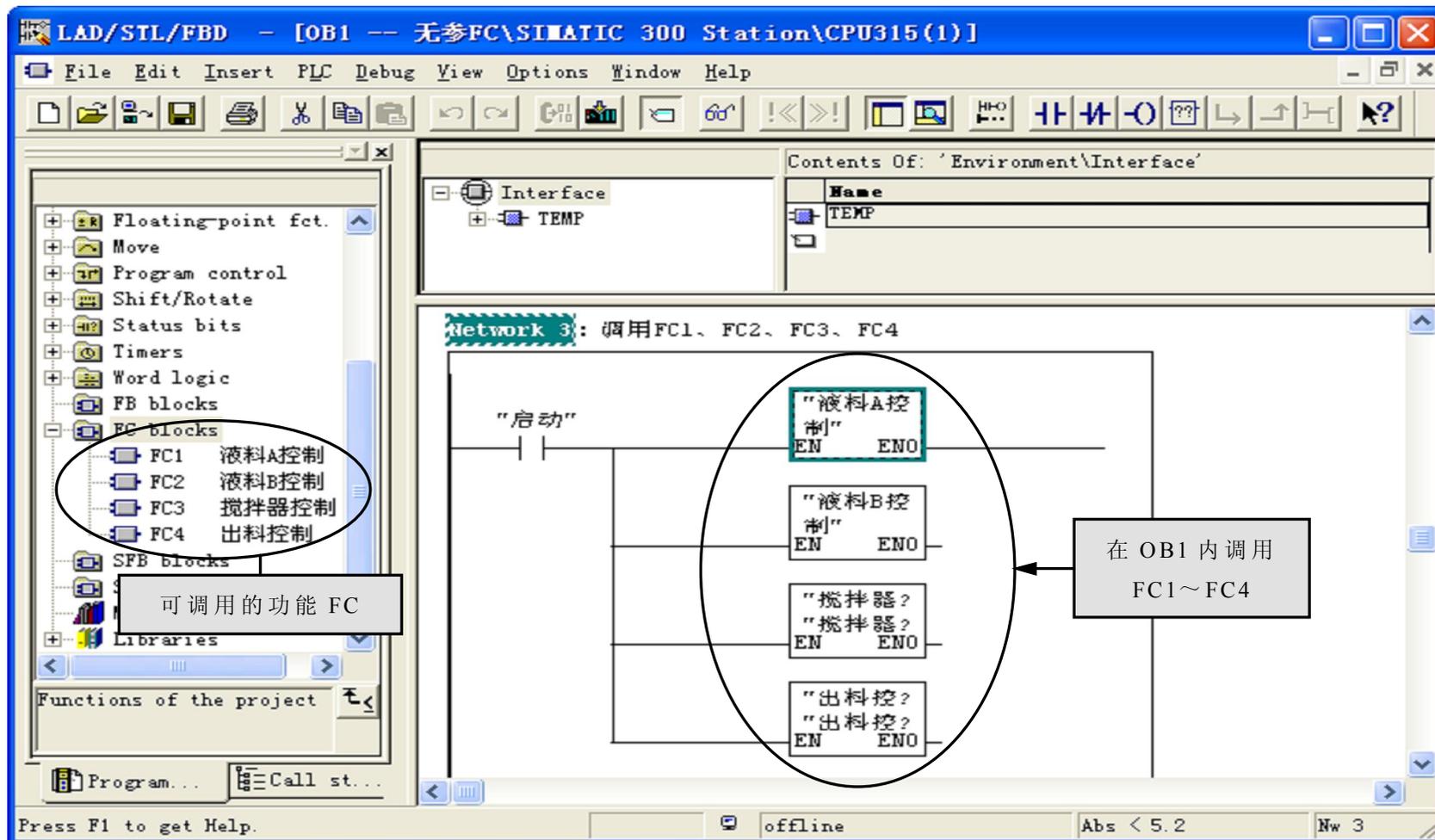
**Network 1:** 初始化所有输出变量



返回上级

# CH5 S7程序结构与程序设计

## § 5.4.2 在OB1中调用无参功能（FC）



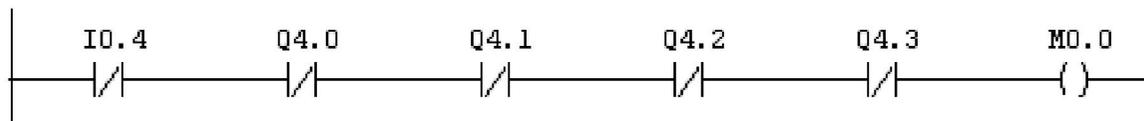
返回本节

# CH5 S7程序结构与程序设计

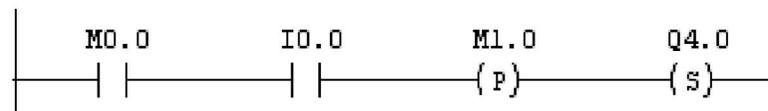
OB1 : “分部式结构的搅拌机控制程序-主循环组织块”

## ➤ OB1的控制程序

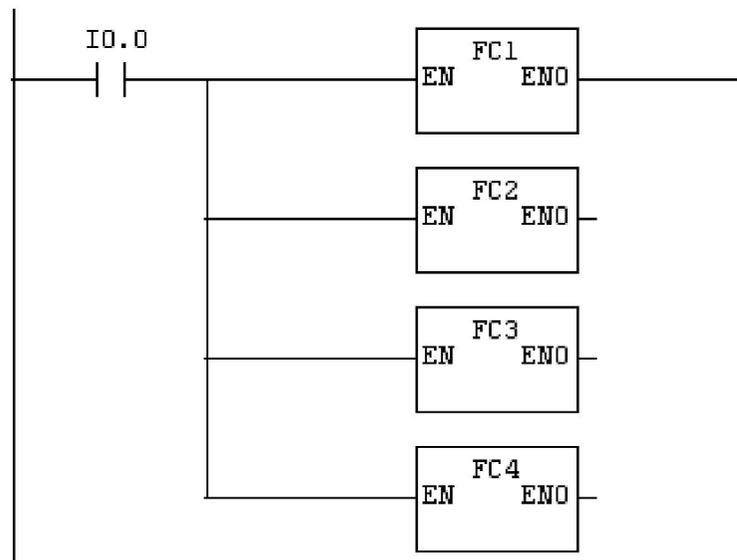
**Network 1:** 设置原始标志



**Network 2:** 启动进料泵1



**Network 3:** 调用FC1、FC2、FC3、FC4



返回上级

## § 5.5 编辑并调用有参功能（FC）

### ——结构化程序设计

所谓**有参功能**（FC），是指编辑功能（FC）时，在局部变量声明表内定义了形式参数，在功能（FC）中使用了**虚拟的符号地址**完成控制程序的编程，以便在其他块中能重复调用有参功能（FC）。这种方式一般应用于结构化程序编写。

#### § 5.5.1 编辑有参功能（FC）

#### § 5.5.2 在OB1中调用有参功能（FC）



## CH5 S7程序结构与程序设计

### § 5.5.1 编辑有参功能（FC）

- 创建多级分频器的S7项目
- 硬件配置
- 编写符号表
- 规划程序结构
- 创建有参功能FC1

返回本节

## CH5 S7程序结构与程序设计

### 1. 创建多级分频器的S7项目

使用菜单【File】→【“New Project”Wizard】创建多级分频器的S7项目，并命名为“有参FC”。

### 2. 硬件配置

打开“SIMATIC 300 Station”文件夹，双击硬件配置图标打开硬件配置窗口，并按图所示完成硬件配置。

Slot	Module	Order number	Fi...	MPI address	I address	Q address	Comment
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU315 (1)	6ES7 315-1AF03-0AB0		2			
3							
4	DI32xDC24V	6ES7 321-1BL80-0AA0			0...3		
5	DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0				4...7	
6							

返回上级

## CH5 S7程序结构与程序设计

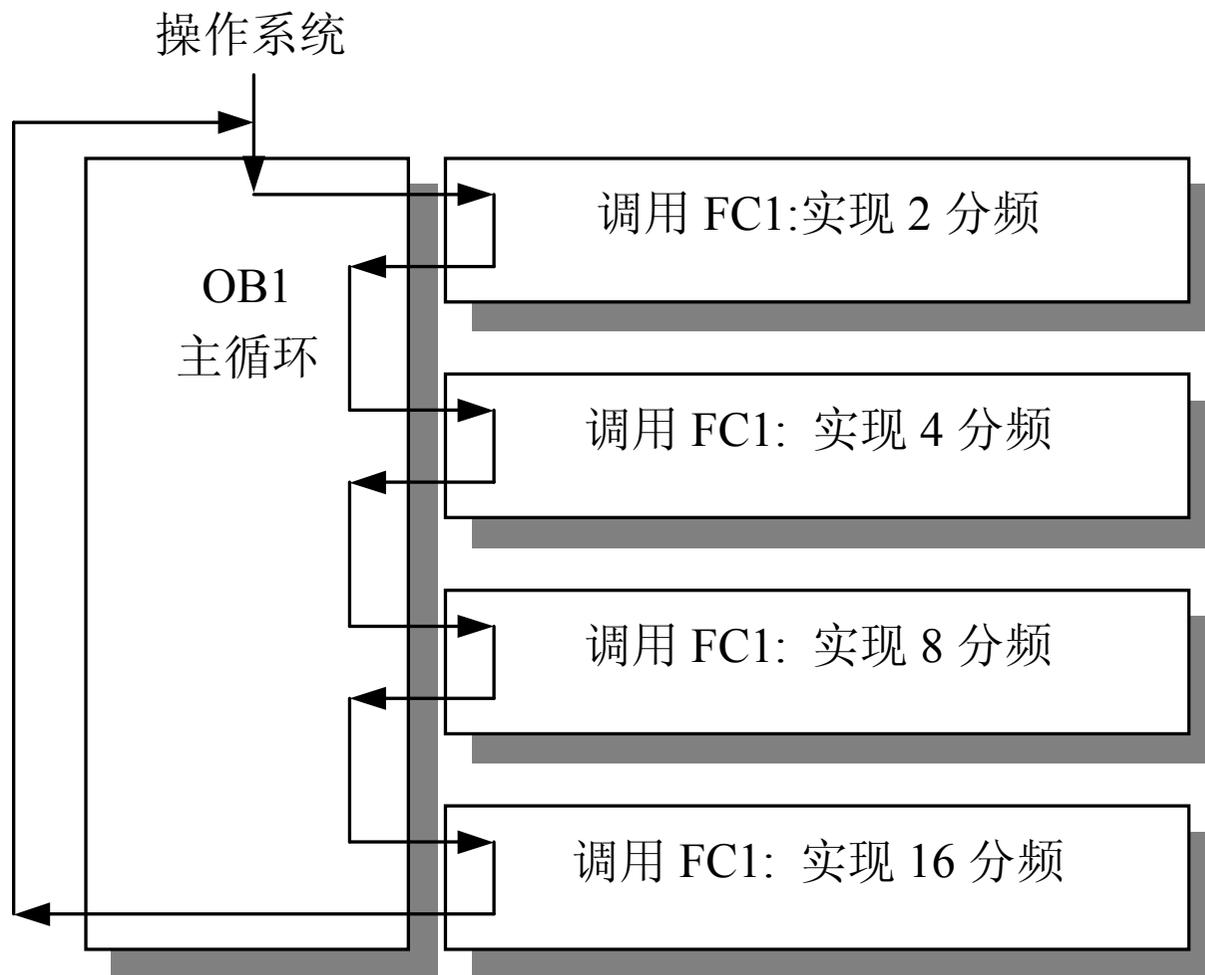
### 3. 编写符号表

	Status	Symbol	Address ▲	Data type	Comment
1		二分频器	FC 1	FC 1	对输入信号二分频
2		In_Port	I 0.0	BOOL	脉冲信号输入端
3		F_P2	M 0.0	BOOL	2分频器上升沿检测标志
4		F_P4	M 0.1	BOOL	4分频器上升沿检测标志
5		F_P8	M 0.2	BOOL	8分频器上升沿检测标志
6		F_P16	M 0.3	BOOL	16分频器上升沿检测标志
7		Cycle Execution	OB 1	OB 1	主循环组织块
8		Out_Port2	Q 4.0	BOOL	2分频器脉冲信号输出端
9		Out_Port4	Q 4.1	BOOL	4分频器脉冲信号输出端
10		Out_Port8	Q 4.2	BOOL	8分频器脉冲信号输出端
11		Out_Port16	Q 4.3	BOOL	16分频器脉冲信号输出端
12		LED2	Q 4.4	BOOL	2分频信号指示灯
13		LED4	Q 4.5	BOOL	4分频信号指示灯
14		LED8	Q 4.6	BOOL	8分频信号指示灯
15		LED16	Q 4.7	BOOL	16分频信号指示灯

返回上级

# CH5 S7程序结构与程序设计

## 4. 规划程序结构



返回上级

### 5. 创建有参FC1

选择“有参FC”项目的“Blocks”文件夹，然后执行菜单命令【Insert】→【S7 Block】→【Function】，在块文件夹内创建一个功能，并命名为“FC1”。

#### ▶ 编辑FC1的变量声明表

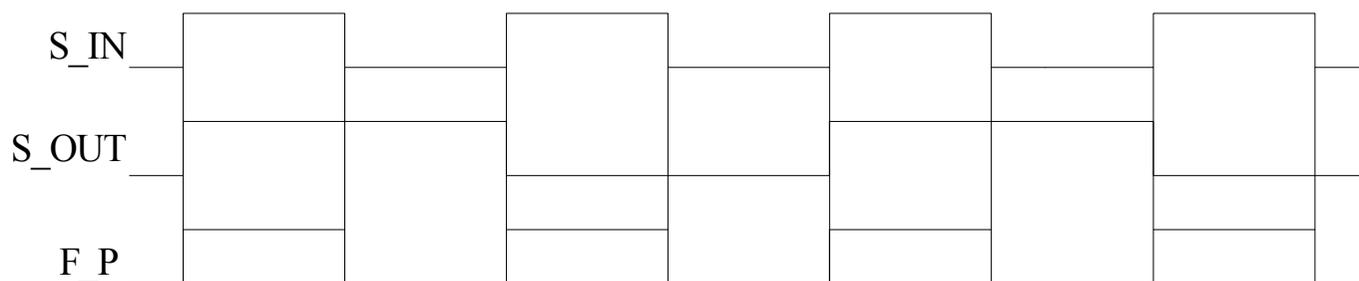
在FC1的变量声明表内，声明4个参数，见下表。

接口类型	变量名	数据类型	注释
In	S_IN	BOOL	脉冲输入信号
Out	S_OUT	BOOL	脉冲输出信号
Out	LED	BOOL	输出状态指示
In_Out	F_P	BOOL	上跳沿检测标志

## CH5 S7程序结构与程序设计

### ▶ 编辑FC1的控制程序

二分频器的时序如图所示。分析二分频器的时序图可以看到，输入信号每出现一个上升沿，输出便改变一次状态，据此可采用上跳沿检测指令实现。

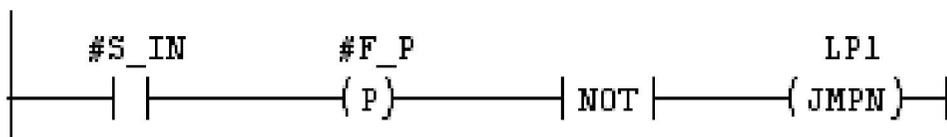


如果输入信号S\_IN出现上升沿，则对S\_OUT取反，然后将S\_OUT的信号状态送LED显示；否则，程序直接跳转到LP1，将S\_OUT的信号状态送LED显示。

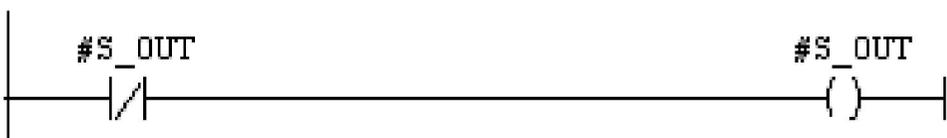
# CH5 S7程序结构与程序设计

FC1 : 二分频程序

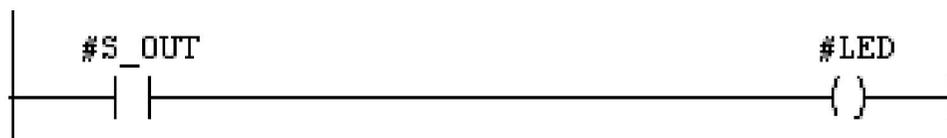
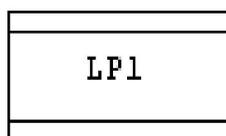
**Network 1**: 二分频程序



**Network 2**: 上升沿检测标志



**Network 3**: Title:



FC1的控制程序

FC1 : 二分频程序

**Network 1**: 二分频程序

```
A      #S_IN
FP     #F_P
NOT
JCN    LP1
```

**Network 2**: 上升沿检测标志

```
AN     #S_OUT
=      #S_OUT
```

**Network 3**: Title:

```
LP1:  A      #S_OUT
      =      #LED
```

返回上级

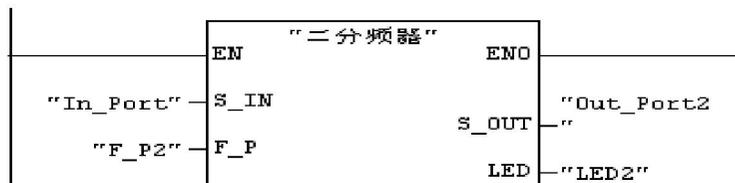
# CH5 S7程序结构与程序设计

## § 5.5.2 在OB1中调用有参功能（FC）

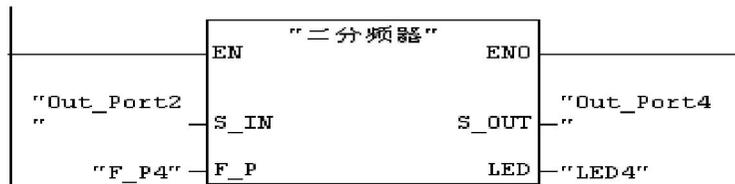
使用符号地址

OB1：“多级分频器主循环组织块，使用符号地址”

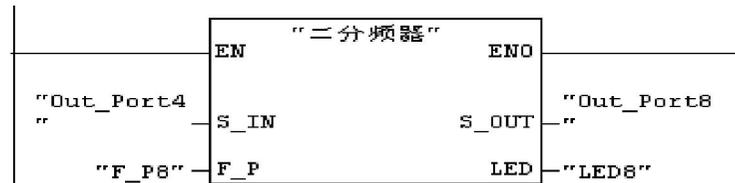
Network 1：调用FC1实现2分频



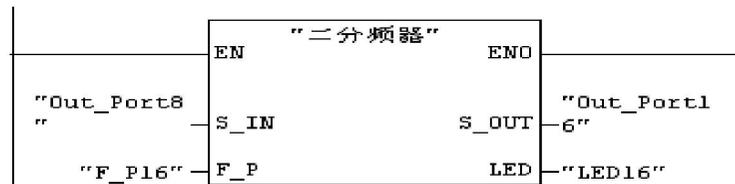
Network 2：调用FC1实现4分频



Network 3：调用FC1实现8分频

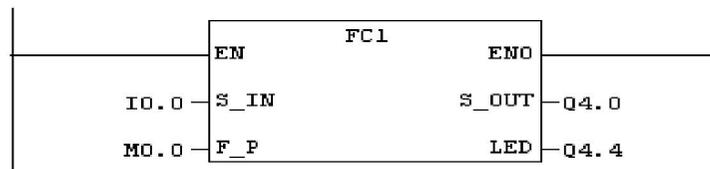


Network 4：调用FC1实现16分频

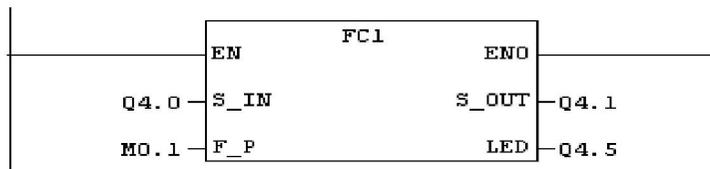


OB1：“多级分频器主循环组织块，使用绝对地址”

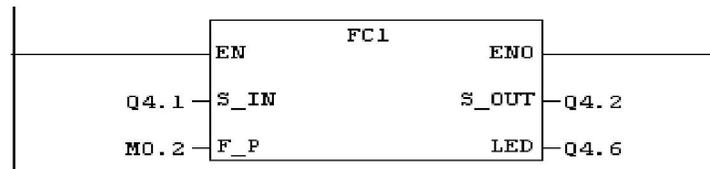
Network 1：调用FC1实现2分频



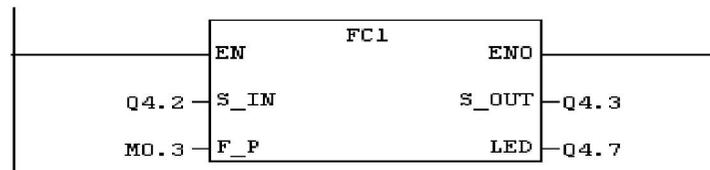
Network 2：调用FC1实现4分频



Network 3：调用FC1实现8分频



Network 4：调用FC1实现16分频



使用绝对地址

## § 5.6 编辑无静参的功能块（FB）

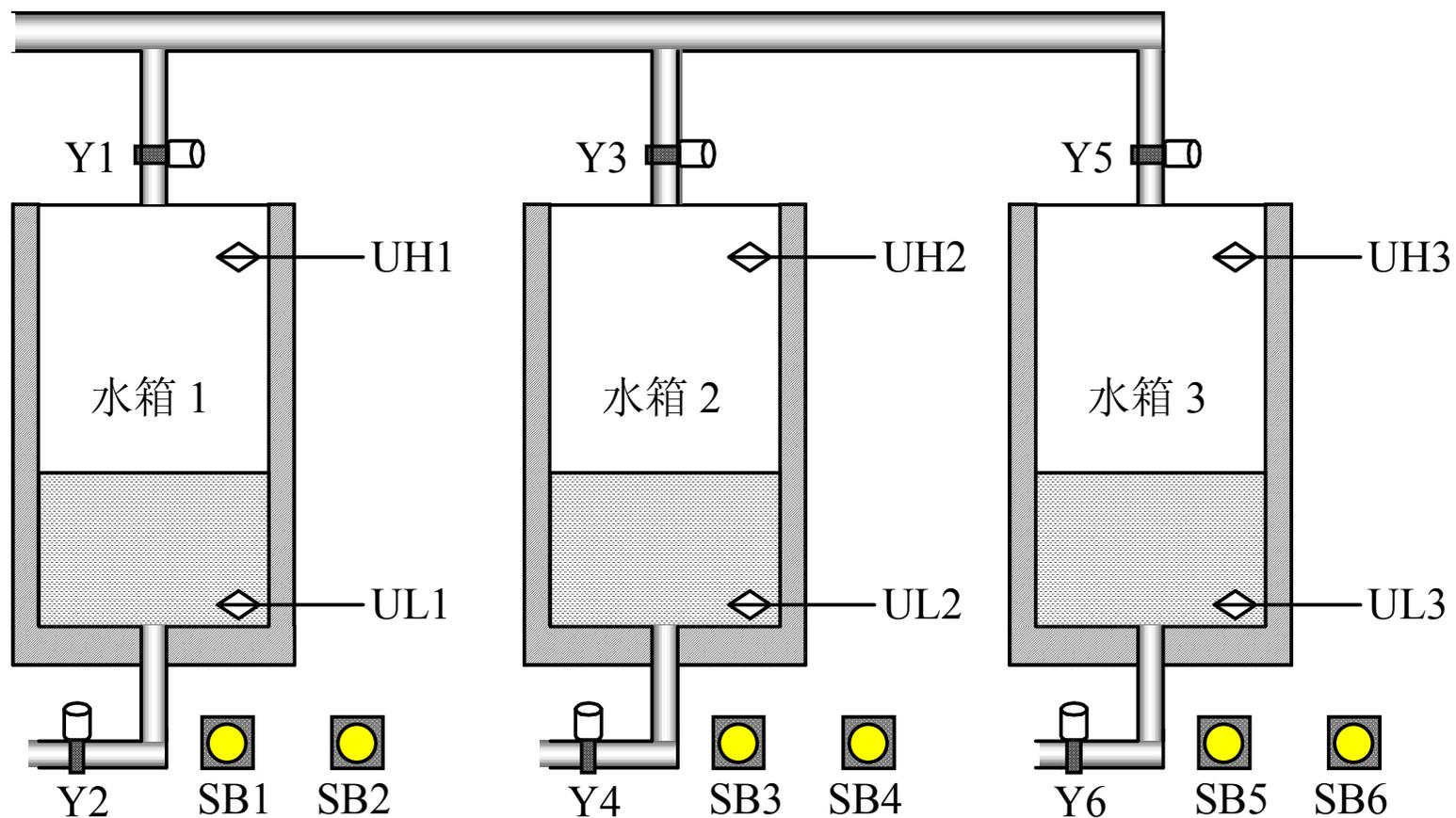
功能块（FB）在程序的体系结构中位于组织块之下。它包含程序的一部分，这部分程序在OB1中可以多次调用。功能块的所有形参和静态数据都存储在一个单独的、被指定给该功能块的数据块（DB）中，该数据块被称为背景数据块。当调用FB时，该背景数据块会自动打开，实际参数的值被存储在背景数据块中；当块退出时，背景数据块中的数据仍然保持。

### § 5.6.1 编辑无静态参数的功能块（FB）

### § 5.6.2 在OB1中调用无静态参数的功能块（FB）

## CH5 S7程序结构与程序设计

### 【例5-6-1】 水箱水位控制系统程序设计。



返回本节

## CH5 S7程序结构与程序设计

### 【例5-6-1】

系统有3个贮水箱，每个水箱有2个液位传感器，UH1，UH2，UH3为高液位传感器，“1”有效；UL1，UL2，UL3为低液位传感器，“0”有效。Y1、Y3、Y5分别为3个贮水水箱进水电磁阀；Y2、Y4、Y6分别为3个贮水水箱放水电磁阀。SB1、SB3、SB5分别为3个贮水水箱放水电磁阀手动开启按钮；SB2、SB4、SB6分别为3个贮水水箱放水电磁阀手动关闭按钮。

**控制要求：**SB1、SB3、SB5在PLC外部操作设定，通过人为的方式，按随机的顺序将水箱放空。只要检测到水箱“空”的信号，系统就自动地向水箱注水，直到检测到水箱“满”信号为止。水箱注水的顺序要与水箱放空的顺序相同，每次只能对一个水箱进行注水操作。

## CH5 S7程序结构与程序设计

### § 5.6.1 编辑无静态参数的功能块（FB）

- 创建S7项目
- 硬件配置
- 编写符号表
- 规划程序结构
- 编辑功能块（FB）
- 建立背景数据块（DI）
- 编辑启动组织块OB100

返回本节

## CH5 S7程序结构与程序设计

### 1. 创建S7项目

使用菜单【File】→【“New Project”Wizard】创建水箱水位控制系统的S7项目，并命名为“无静参FB”。项目包含组织块OB1和OB100。

### 2. 硬件配置

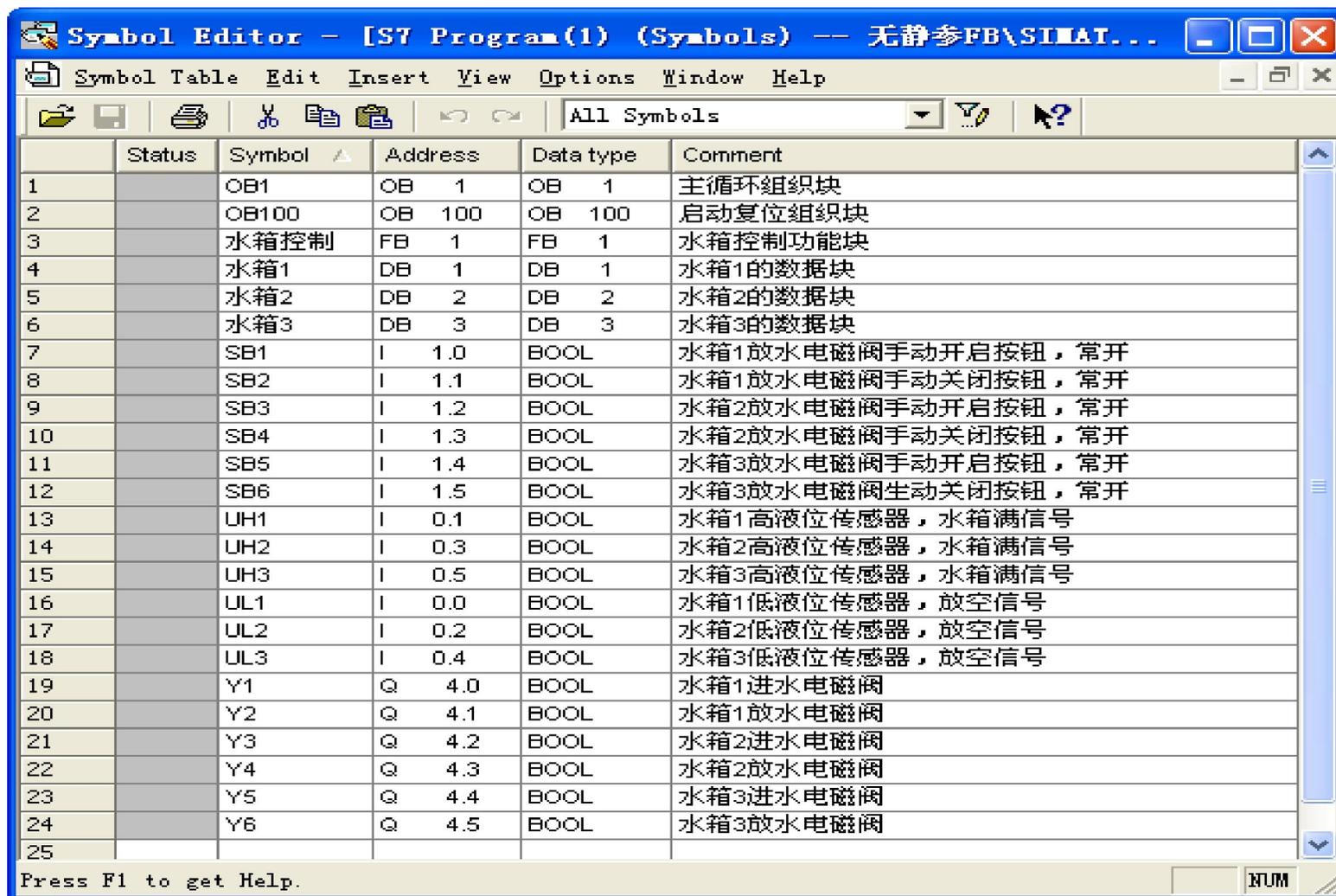
在“无静参FB”项目内打开“SIMATIC 300 Station”文件夹，打开硬件配置窗口，并按图所示完成硬件配置。

Slot	Module	Order number	Fi...	MPI address	I address	Q address	Comment
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU315 (1)	6ES7 315-1AF03-0AB0		2			
3							
4	DI32xDC24V	6ES7 321-1BL80-0AA0			0...3		
5	DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0				4...7	
6							

返回上级

# CH5 S7程序结构与程序设计

## 3. 编写符号表



Symbol Editor - [S7 Program(1) (Symbols) -- 无静参FB\SIMAT...

Symbol Table Edit Insert View Options Window Help

All Symbols

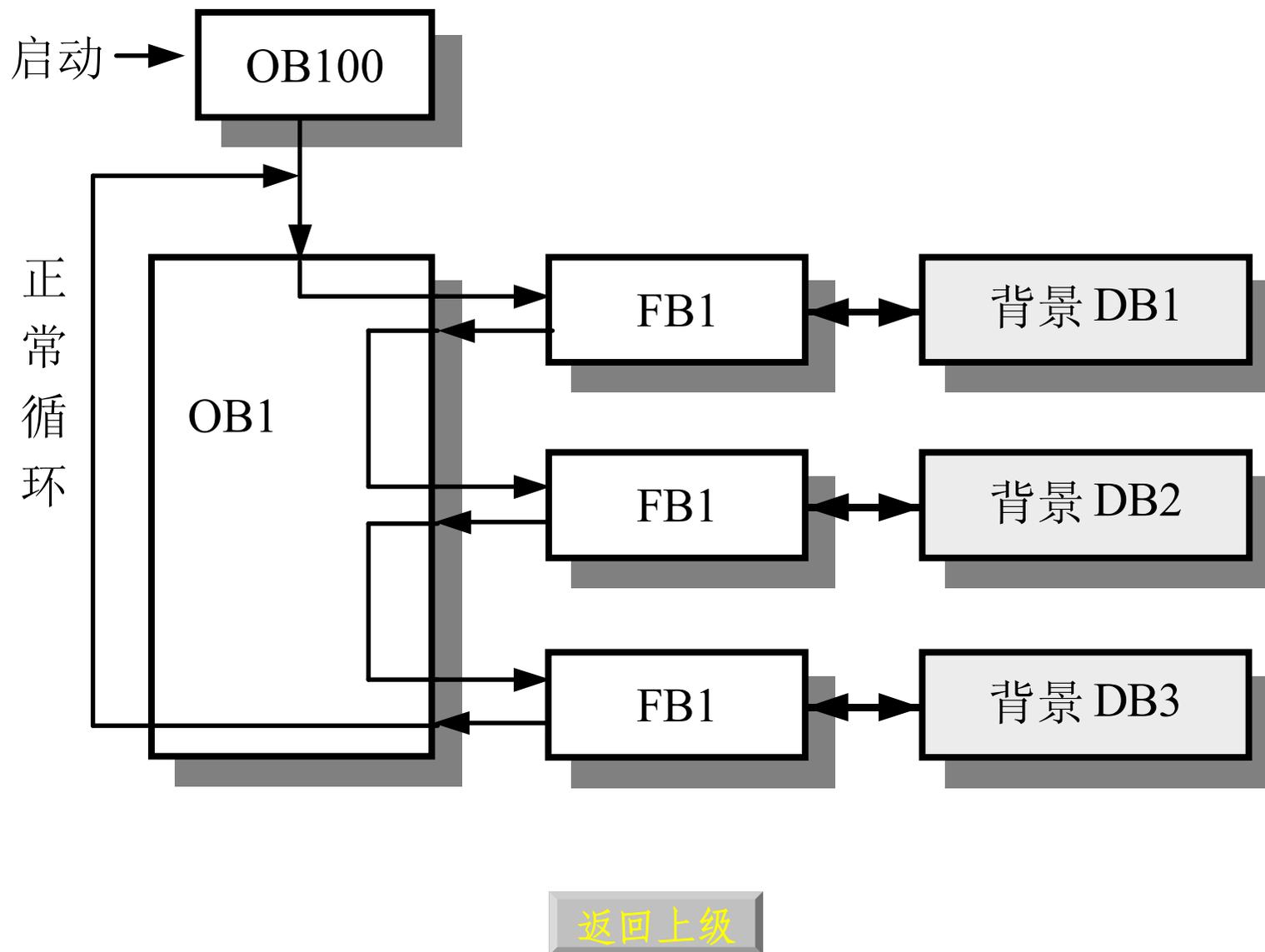
	Status	Symbol ▲	Address	Data type	Comment
1		OB1	OB 1	OB 1	主循环组织块
2		OB100	OB 100	OB 100	启动复位组织块
3		水箱控制	FB 1	FB 1	水箱控制功能块
4		水箱1	DB 1	DB 1	水箱1的数据块
5		水箱2	DB 2	DB 2	水箱2的数据块
6		水箱3	DB 3	DB 3	水箱3的数据块
7		SB1	I 1.0	BOOL	水箱1放水电磁阀手动开启按钮，常开
8		SB2	I 1.1	BOOL	水箱1放水电磁阀手动关闭按钮，常开
9		SB3	I 1.2	BOOL	水箱2放水电磁阀手动开启按钮，常开
10		SB4	I 1.3	BOOL	水箱2放水电磁阀手动关闭按钮，常开
11		SB5	I 1.4	BOOL	水箱3放水电磁阀手动开启按钮，常开
12		SB6	I 1.5	BOOL	水箱3放水电磁阀手动关闭按钮，常开
13		UH1	I 0.1	BOOL	水箱1高液位传感器，水箱满信号
14		UH2	I 0.3	BOOL	水箱2高液位传感器，水箱满信号
15		UH3	I 0.5	BOOL	水箱3高液位传感器，水箱满信号
16		UL1	I 0.0	BOOL	水箱1低液位传感器，放空信号
17		UL2	I 0.2	BOOL	水箱2低液位传感器，放空信号
18		UL3	I 0.4	BOOL	水箱3低液位传感器，放空信号
19		Y1	Q 4.0	BOOL	水箱1进水电磁阀
20		Y2	Q 4.1	BOOL	水箱1放水电磁阀
21		Y3	Q 4.2	BOOL	水箱2进水电磁阀
22		Y4	Q 4.3	BOOL	水箱2放水电磁阀
23		Y5	Q 4.4	BOOL	水箱3进水电磁阀
24		Y6	Q 4.5	BOOL	水箱3放水电磁阀
25					

Press F1 to get Help. NUM

返回上级

# CH5 S7程序结构与程序设计

## 4. 规划程序结构



### 5. 编辑功能（FB1）

在“**无静参FB**”项目内选择“**Blocks**”文件夹，执行菜单命令【Insert】→【S7 Block】→【Function Block】，创建功能块FB1。由于在符号表内已经为FB1定义了符号名，因此在FB1的属性对话框内系统会自动添加符号名“**水箱控制**”。

- ✓ 定义局部变量声明表
- ✓ 编写程序代码

## CH5 S7程序结构与程序设计

### ▶ 定义局部变量声明表

与功能（FC）不同，在功能块（FB）参数表内还有扩展地址（Exclusion address）和结束地址（Termination address）选项。

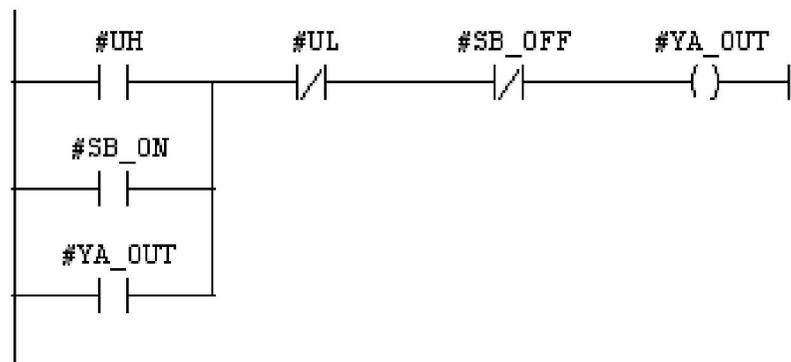
接口类型	变量名	数据类型	地址	初始值	扩展地址	结束地址	注释
In	UH	BOOL	0.0	FALSE	-	-	高液位传感器，表示水箱满
	UL	BOOL	0.1	FALSE	-	-	低液位传感器，表示水箱空
	SB_ON	BOOL	0.2	FALSE	-	-	放水电磁阀开启按钮，常开
	SB_OFF	BOOL	0.3	FALSE	-	-	放水电磁阀关闭按钮，常开
	B_F	BOOL	0.4	FALSE	-	-	水箱 B 空标志
	C_F	BOOL	0.5	FALSE	-	-	水箱 C 空标志
	YB_IN	BOOL	0.6	FALSE	-	-	水箱 B 进水电磁阀
	YC_IN	BOOL	0.7	FALSE	-	-	水箱 C 进水电磁阀
Out	YA_IN	BOOL	2.0	FALSE	-	-	当前水箱 A 进水电磁阀
	YA_OUT	BOOL	2.1	FALSE	-	-	当前水箱 A 放水电磁阀
	A_F	BOOL	2.2	FALSE	-	-	当前水箱（A）空标志

返回上级

# CH5 S7程序结构与程序设计

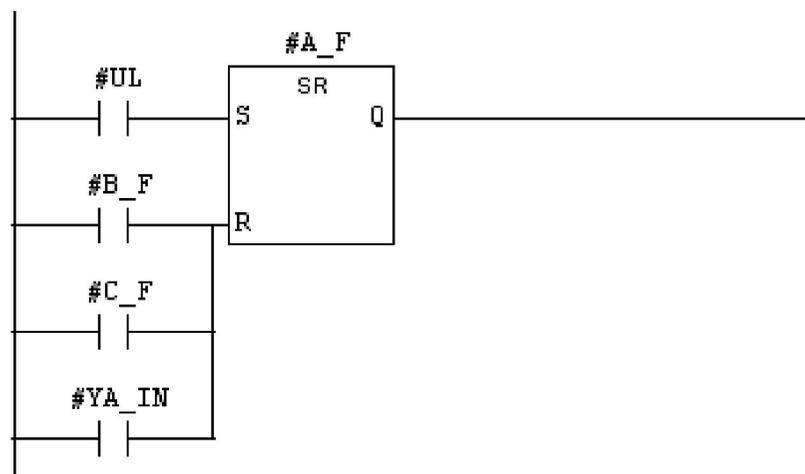
FB1 : 水箱控制

**Network 1**: 水箱放水控制

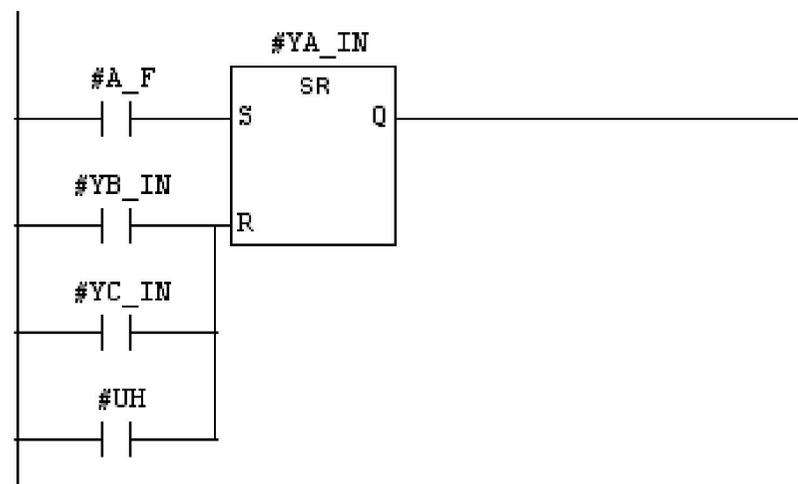


编写程序代码

**Network 2**: 设置水箱空标志



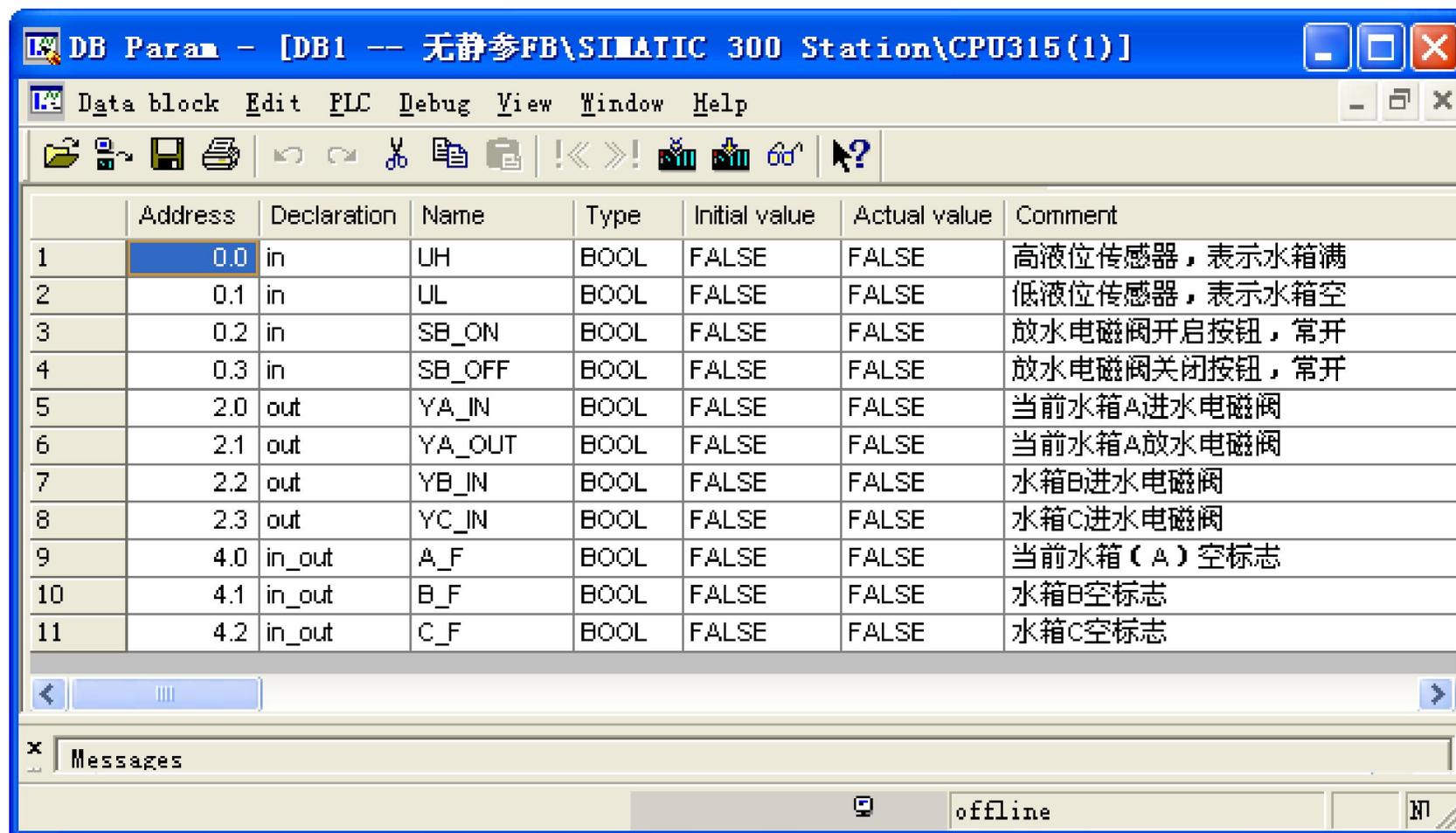
**Network 3**: 水箱进水控制



返回上级

## CH5 S7程序结构与程序设计

### 6. 建立背景数据块DB1、DB2、DB3



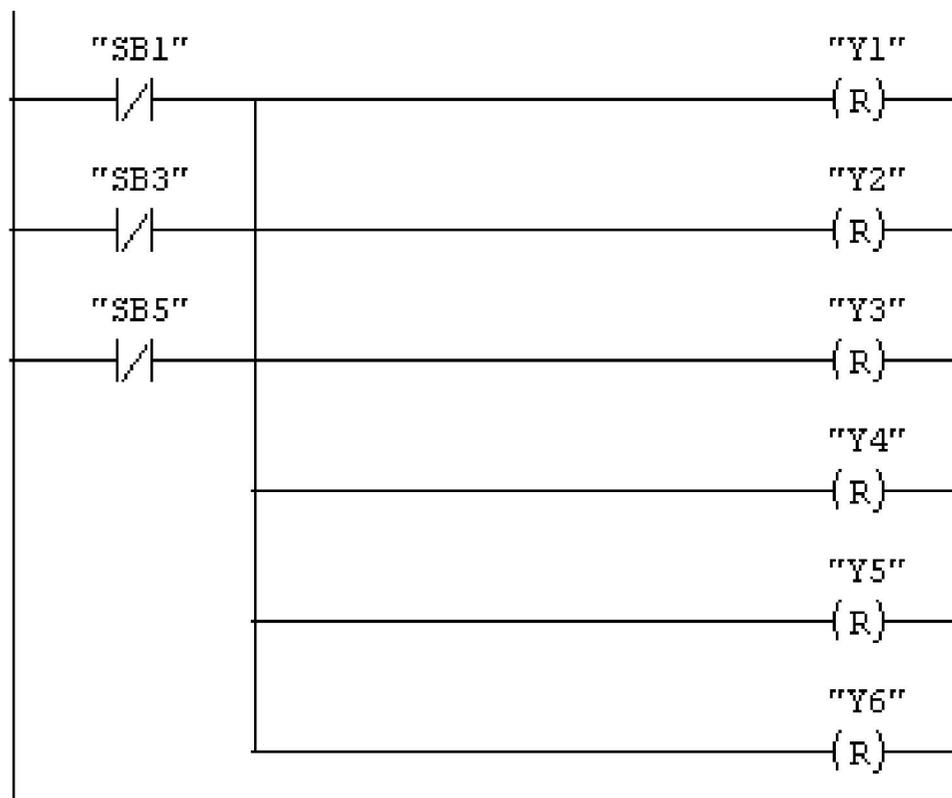
返回上级

## CH5 S7程序结构与程序设计

### 7. 编辑启动组织块OB100

OB100 : "Complete Restart"

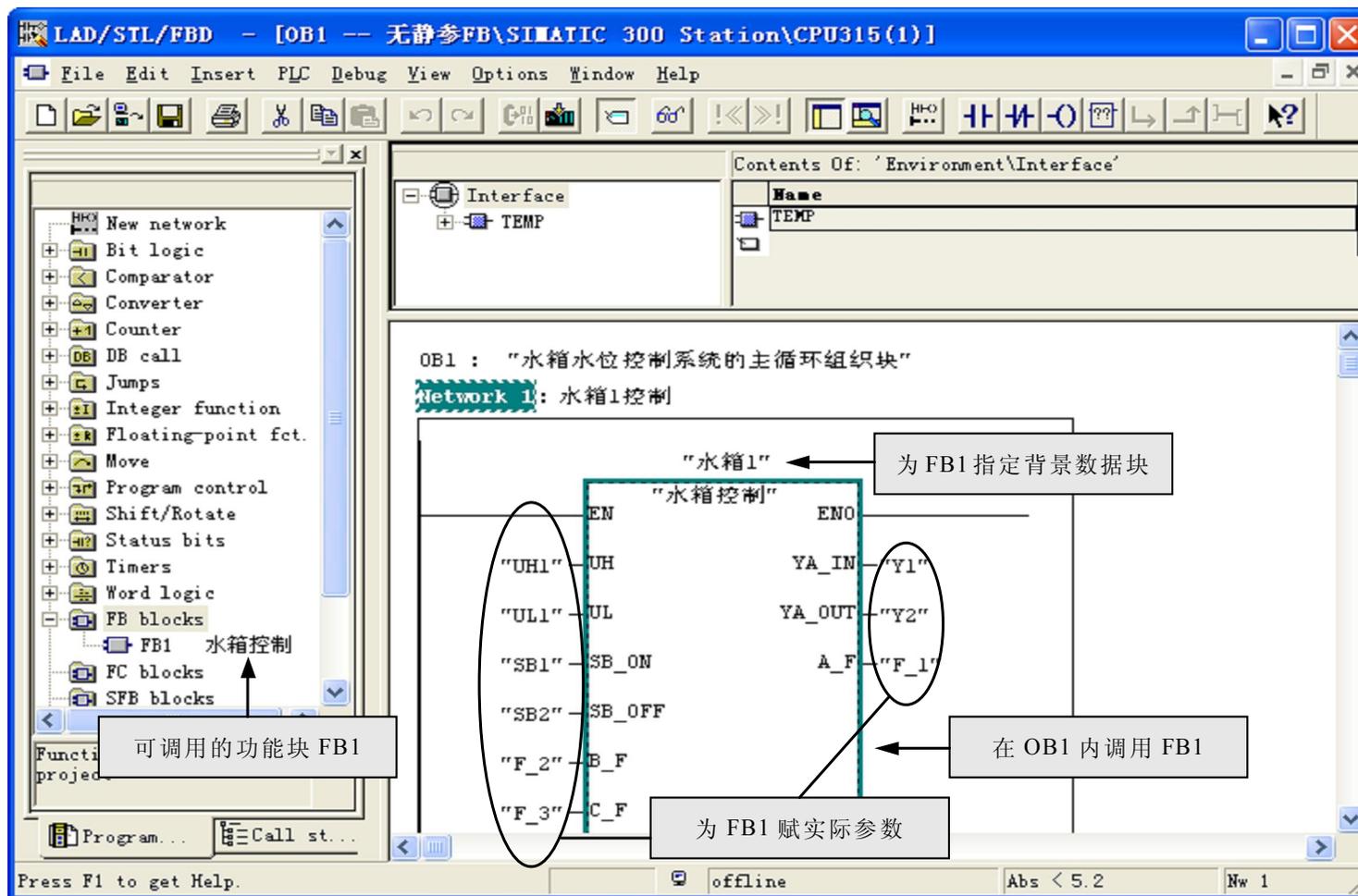
**Network 1**: 对电磁阀复位



返回上级

# CH5 S7程序结构与程序设计

## § 5.6.2 在OB1中调用无静态参数的功能块（FB）

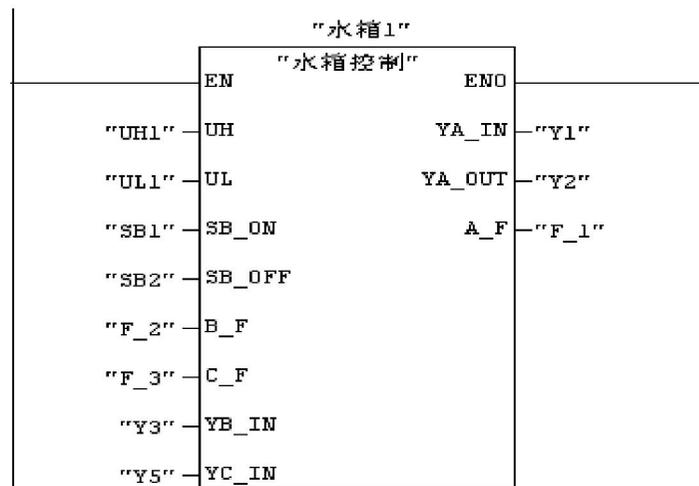


返回本节

# CH5 S7程序结构与程序设计

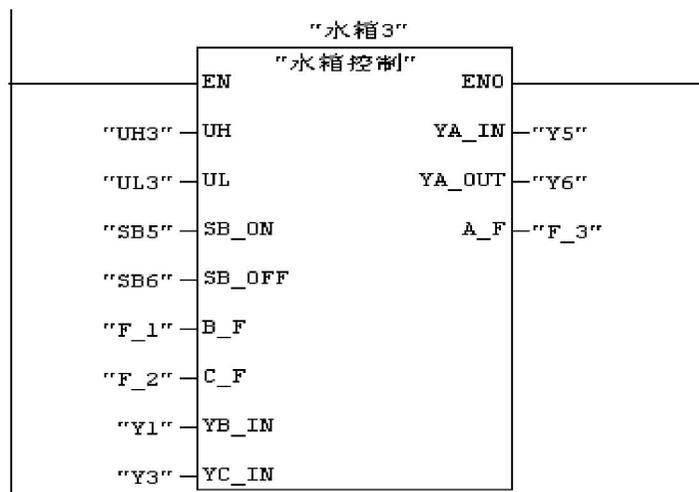
OB1 : “水箱水位控制系统的主循环组织块”

**Network 1**: 水箱1控制

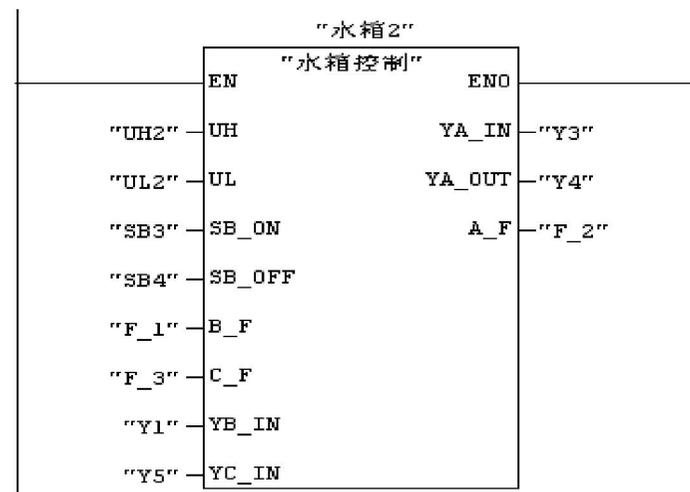


## OB1控制程序

**Network 3**: 水箱3控制



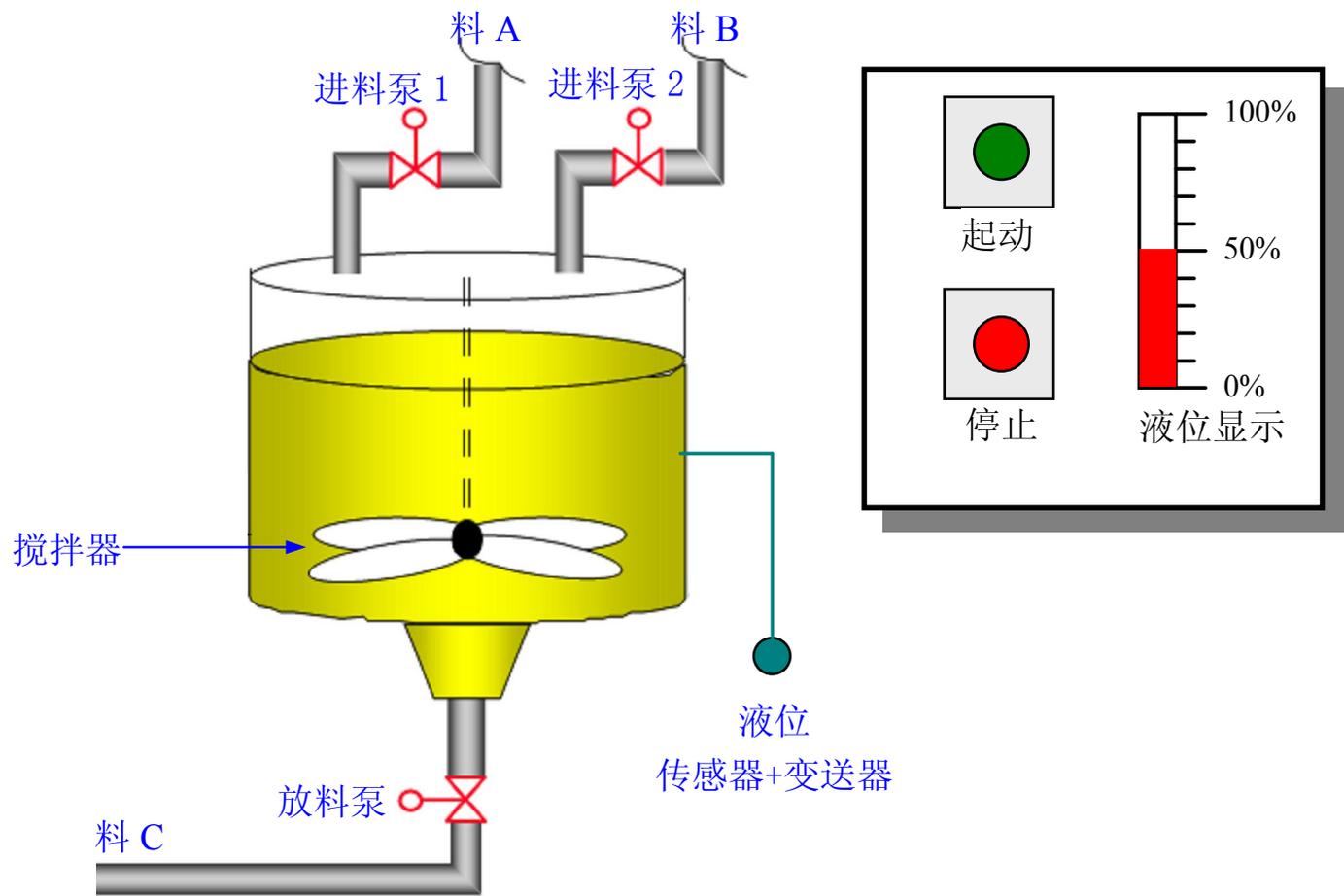
**Network 2**: 水箱2控制



返回上级

# CH5 S7程序结构与程序设计

## 【例5-6-2】 搅拌控制系统程序设计——使用模拟量。



返回上级

## CH5 S7程序结构与程序设计

上图所示为一**搅拌控制系统**，由一个模拟量液位传感器-变送器来检测液位的高低，并进行液位显示。现要求对A、B两种液体原料按等比例混合，请编写控制程序，控制要求如下：

按起动按钮后系统自动运行，首先打开进料泵1，开始加入液料A→当液位达到50%后，则关闭进料泵1，打开进料泵2，开始加入液料B→当液位达到100%后，则关闭进料泵2，起动搅拌器→搅拌10s后，关闭搅拌器，开启放料泵→当液料放空后，延时5s后关闭放料泵。按停止按钮，系统应立即停止运行。

[返回上级](#)

## CH5 S7程序结构与程序设计

### 1. 创建S7项目

使用菜单【File】→【“New Project”Wizard】创建搅拌控制系统的S7项目，并命名为“FC与FB”。

### 2. 硬件配置

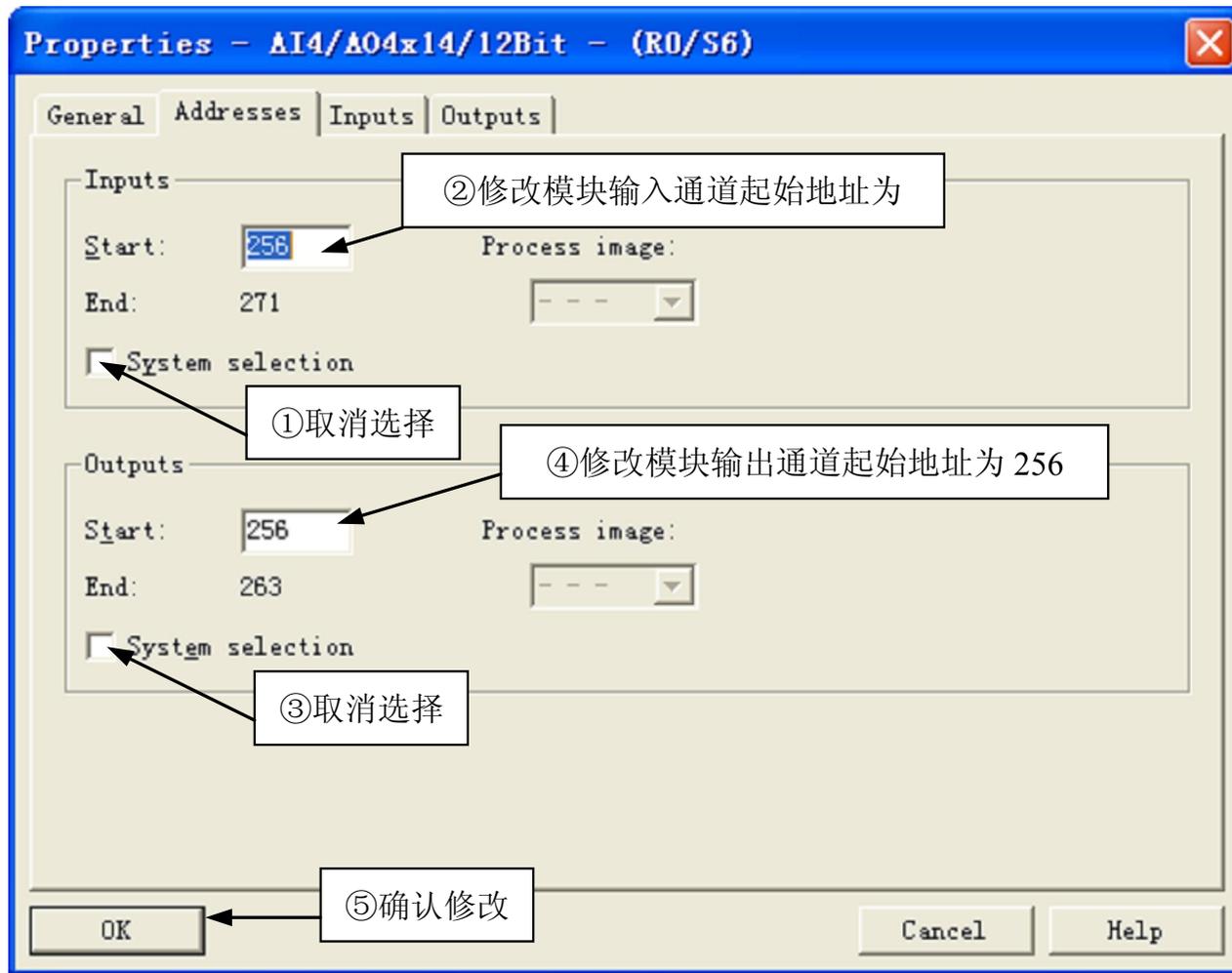
在“FC与FB”项目内打开“SIMATIC 300 Station”文件夹，打开硬件配置窗口，并按图下完成硬件配置。

S...	Module	Order number	Firmware	MPI address	I add...	Q add...	C...
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU 315	6ES7 315-1AF03-0AB0	V1.2	2			
3							
4	DI32xDC24V	6ES7 321-1BL80-0AA0			0...3		
5	DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0				4...7	
6	AI4/AO4x14/12Bit	6ES7 335-7HG01-0AB0			256...271	256...263	
7							
8							

返回上级

# CH5 S7程序结构与程序设计

修改模块的模拟量输入通道和输出通道的起始地址均为 256。



返回上级

## CH5 S7程序结构与程序设计

### 3. 编辑符号表

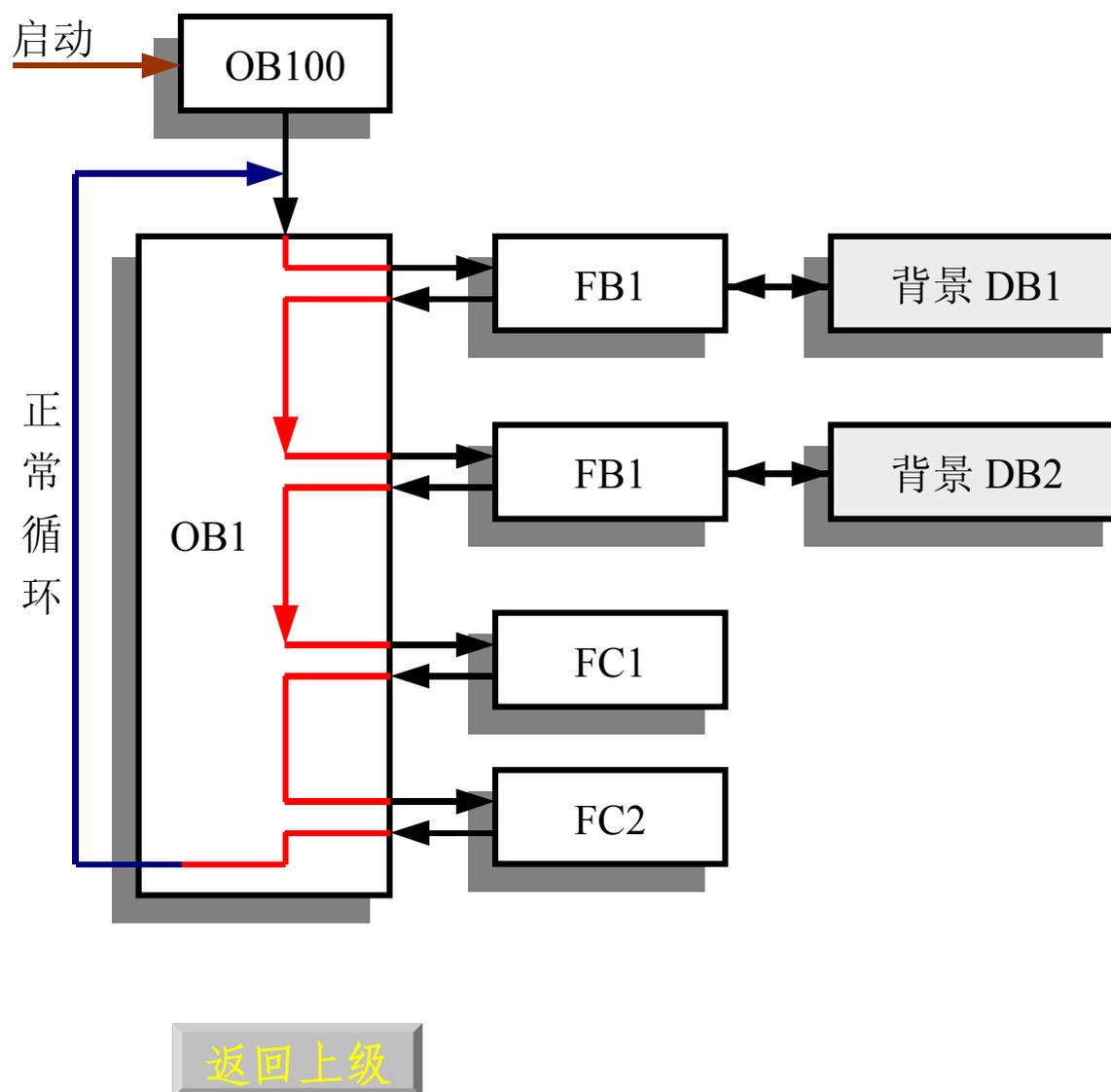
	Statu	Symbol	Address	Data typ	Comment
1		进料控制	FB 1	FB 1	液料 A和液料B进料控制, 进料结束起动手动搅拌器
2		搅拌控制	FC 1	FC 1	搅拌器延时关闭, 并启动放料泵
3		放料控制	FC 2	FC 2	比较最低液位, 控制延时放空并复位放料泵
4		启动	I 0.0	BOOL	起动手按钮, 常开 (带自锁)
5		停止	I 0.1	BOOL	停止按钮, 常开
6		原始标志	M 0.0	BOOL	表示进料泵、放料泵及搅拌器均处于停机?..
7		最低液位标志	M 0.1	BOOL	表示液料即将放空
8		Cycle Execution	OB 1	OB 1	线性结构的搅拌器控制程序
9		BS	PIW 256	WORD	液位传感器-变送器, 送出模拟量液位信号
10		DISP	PQW 256	WORD	液位指针式显示器, 接收模拟量液位信号
11		进料泵1	Q 4.0	BOOL	"1" 有效
12		进料泵2	Q 4.1	BOOL	"1" 有效
13		搅拌器M	Q 4.2	BOOL	"1" 有效
14		放料泵	Q 4.3	BOOL	"1" 有效
15		搅拌定时器	T 1	TIMER	SD定时器, 搅拌10s
16		排空定时器	T 2	TIMER	SD定时器, 延时5s

返回上级

## CH5 S7程序结构与程序设计

### 4. 规划程序结构

OB1为主循环组织块；OB100为启动组织块；FC1实现搅拌控制；FC2实现放料控制；FB1通过调用DB1和DB2实现液料A和液料B的进料控制；DB1和DB2为液料A和液料B进料控制的背景数据块，在调用FB1时为FB1提供实际参数，并保存过程结果。



# CH5 S7程序结构与程序设计

## 5. 创建无参功能（FC1、FC2）（1/2）

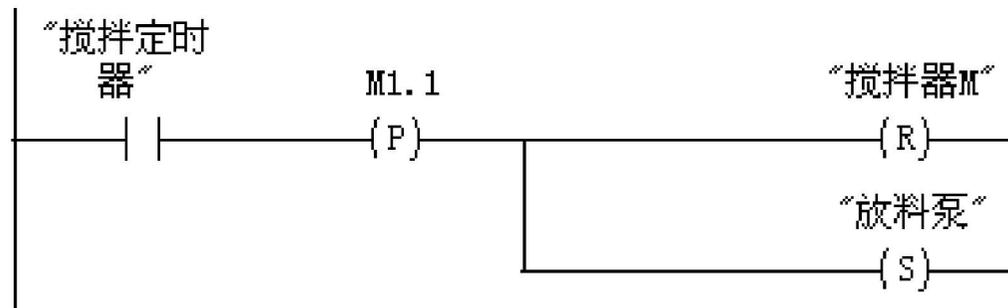
FC1：搅拌器控制

### FC1控制程序

Network 1：搅拌延时



Network 2：关闭搅拌器，启动放料泵



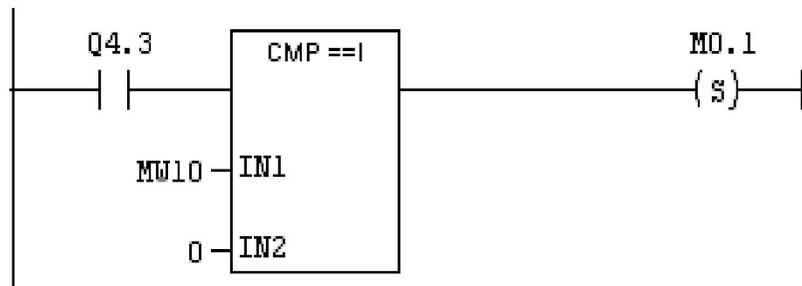
返回上级

# CH5 S7程序结构与程序设计

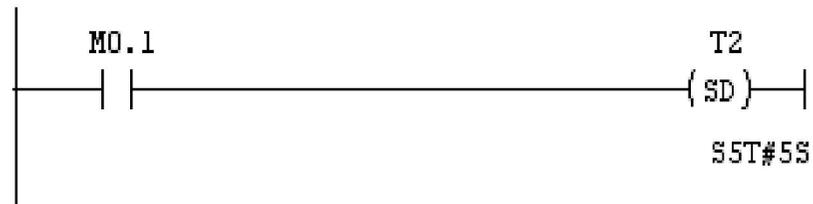
## 5. 创建无参功能（FC1、FC2）（2/2）

FC2：放料控制

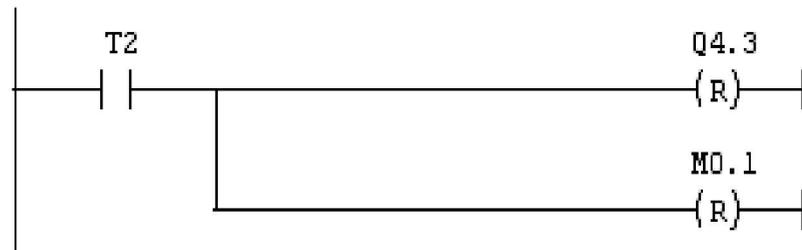
Network 1：设置最低液位标志



Network 2：设置放料延时



Network 3：关闭放料泵，清除最低液位标志



FC2控制程序

## CH5 S7程序结构与程序设计

### 6. 创建无静态参数的功能块（FB1）

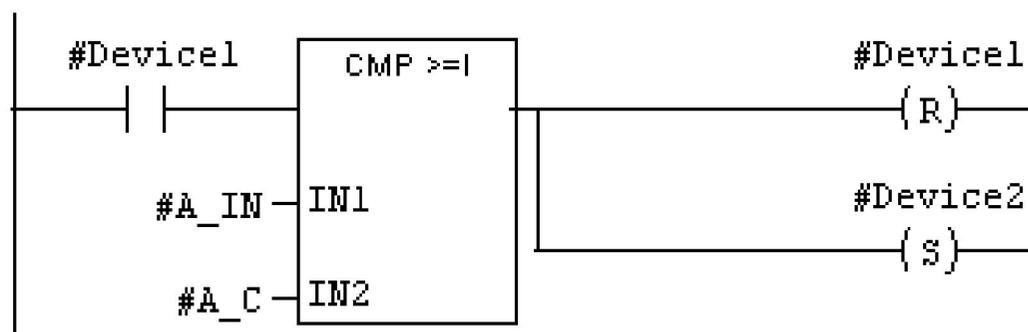
#### ➤ 定义FB1的局部变量声明表

接口类型	变量名	数据类型	地址	初始值	扩展地址	结束地址	注释
In	A_IN	INT	0.0	0	-	-	模拟量输入数据
	A_C	INT	2.0	0	-	-	液位比较值
IN_Out	Device1	BOOL	4.0	FALSE	-	-	设备1
	Device2	BOOL	4.1	FALSE	-	-	设备2

#### ➤ 编写FB1控制程序

FB1：进料控制

Network 1：满足条件，则复位设备1，启动设备2



返回上级

## CH5 S7程序结构与程序设计

### 7. 建立背景数据块（DB1、DB2）

在“FC与FB”项目内选择“Blocks”文件夹，执行菜单命令【Insert】→【S7 Block】→【Data Block】，创建与FB1相关联的背景数据块DB1和DB2。STEP 7自动为DB1和DB2构建了与FB1完全相同的数据结构，如下图所示。

	Address	Declaration	Name	Type	Initial value	Actual value	Comment
1	0.0	in	A_IN	INT	0	0	模拟量输入数据
2	2.0	in	A_C	INT	0	0	比较值
3	4.0	in_out	Device1	BOOL	FALSE	FALSE	设备1
4	4.1	in_out	Device2	BOOL	FALSE	FALSE	设备2

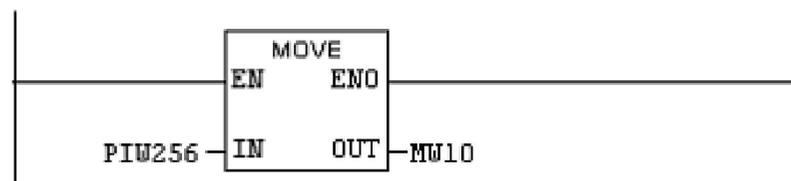
返回上级

# CH5 S7程序结构与程序设计

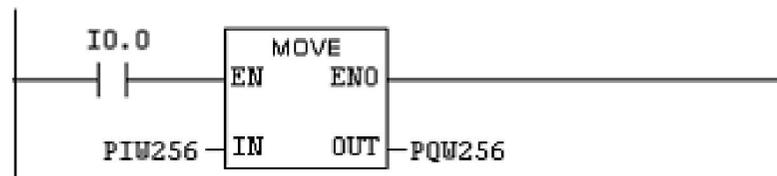
## 8. 在OB1中调用FC1、FC2和FB1

OB1：“搅拌机结构化控制程序-主循环组织块”

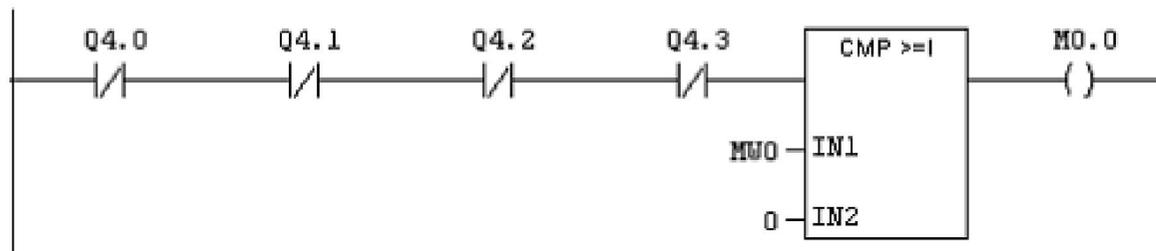
**Network 1:** 设置当前液位信号暂存器



**Network 2:** 将当前液位送显示器显示

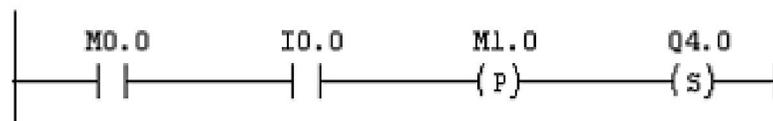


**Network 3:** 设置原始标志

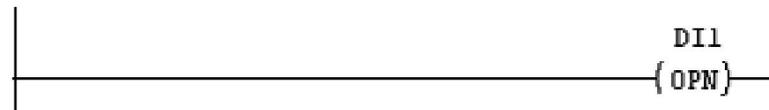


OB1的控制程序

**Network 4:** 启动进料泵1



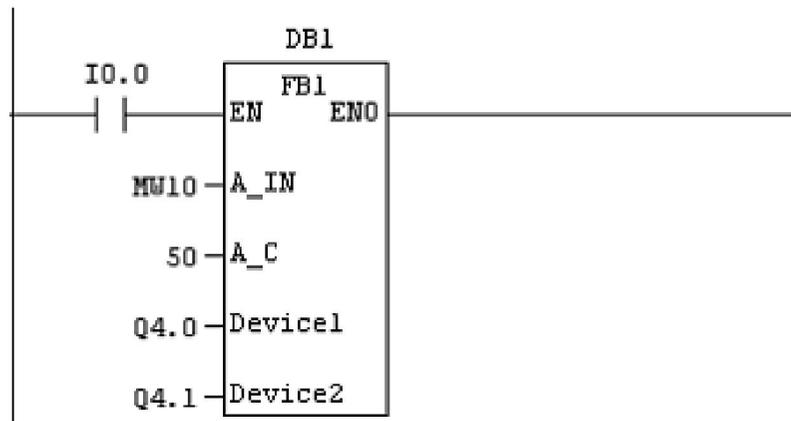
**Network 5:** 打开功能块FB1的背景数据块DB1



返回上级

# CH5 S7程序结构与程序设计

Network 6 : 调用功能块FB1

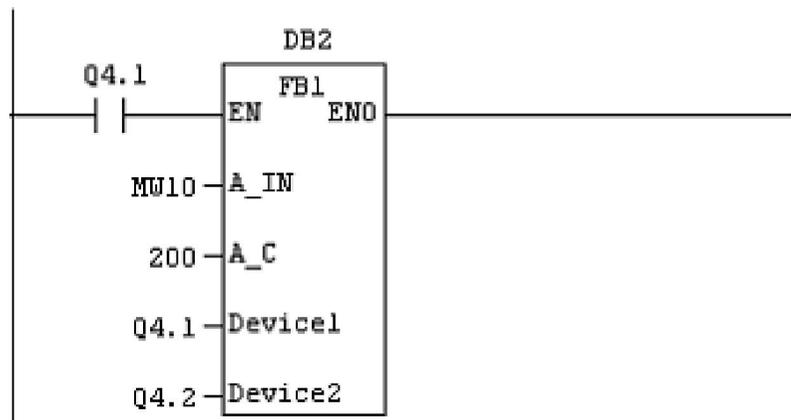


Network 7 : 打开功能块FB1的背景数据块DB2

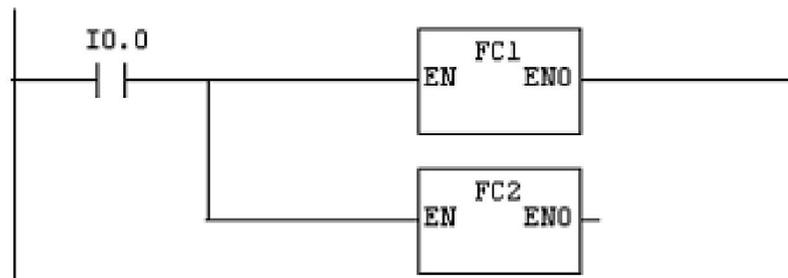


## OB1的控制程序

Network 8 : 调用功能块FB1



Network 9 : 调用功能FC1和FC2

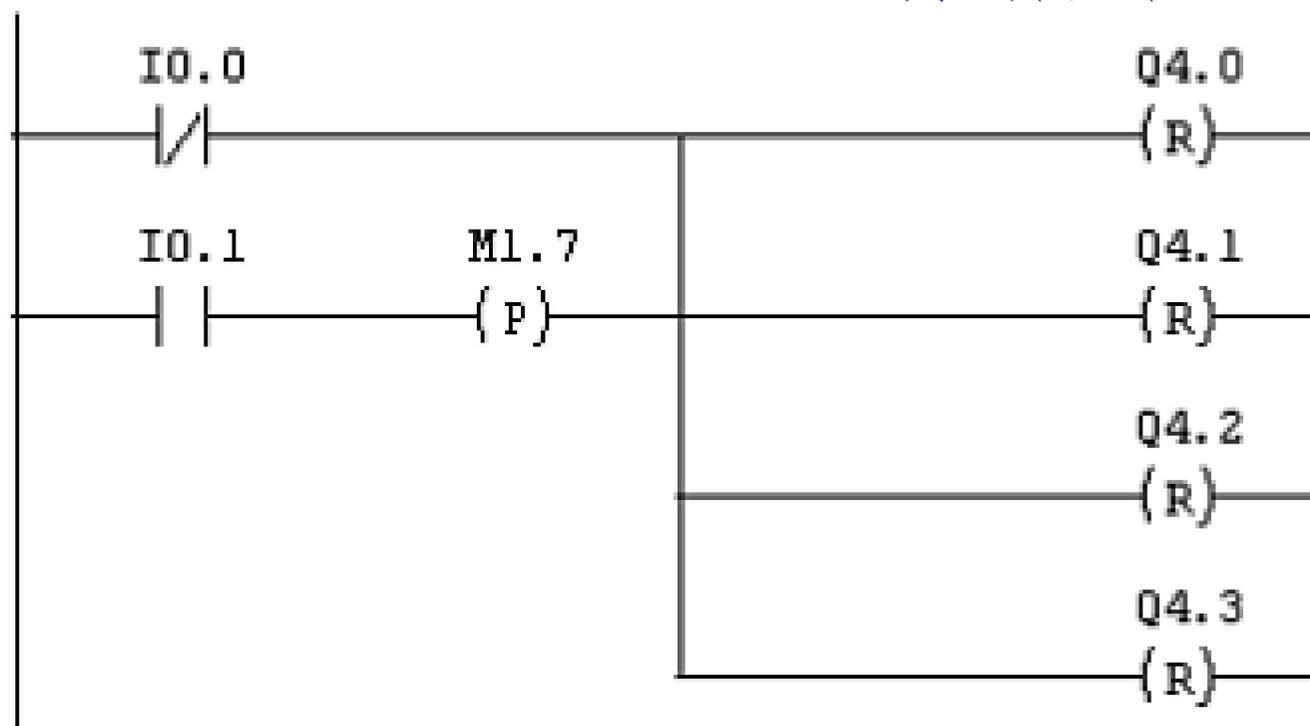


返回上级

## CH5 S7程序结构与程序设计

Network 10 : 复位

OB1的控制程序



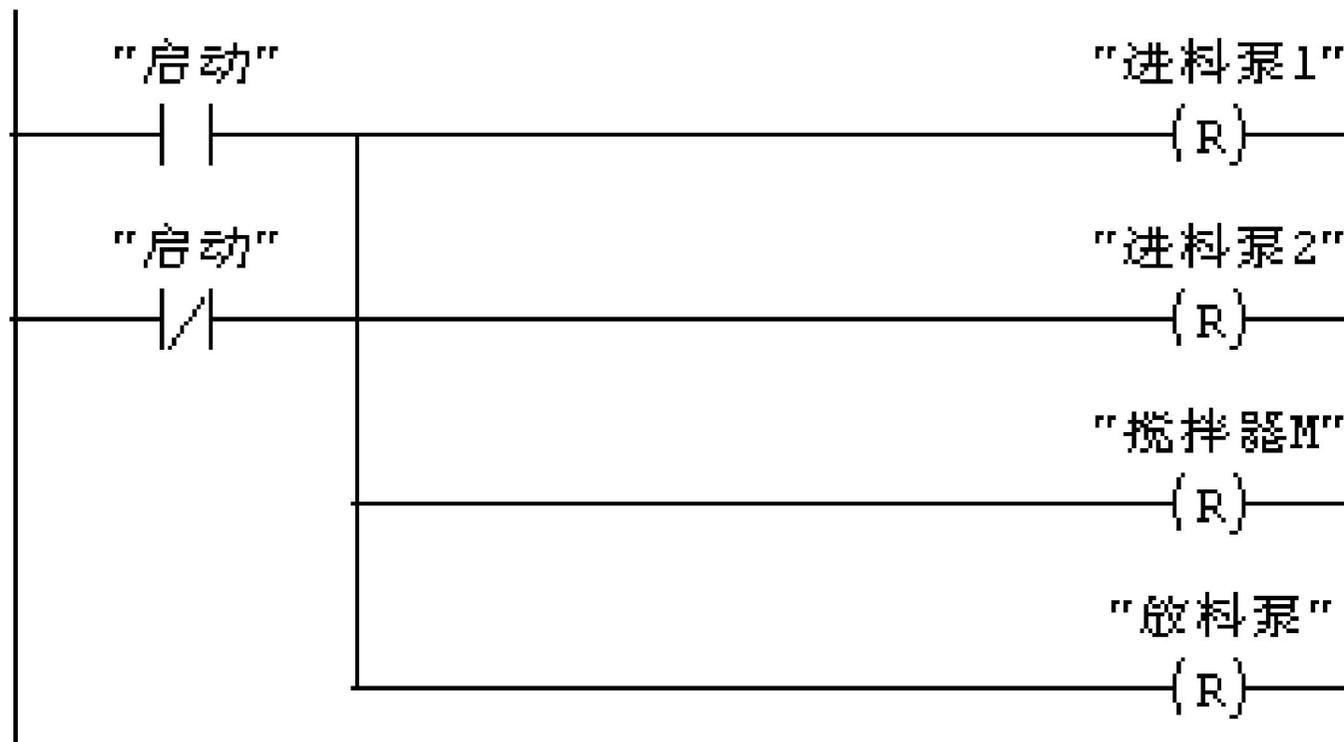
返回上级

## CH5 S7程序结构与程序设计

### 9. 编写启动组织块OB100的控制程序

OB100 : "Complete Restart"

**Network 1**: 复位



返回上级

### § 5.7 编辑并调用有静态参数的功能块

在编辑功能块（FB）时，如果程序中需要特定数据的参数，可以考虑将该特定数据定义为**静态参数**，并在FB的声明表内STAT处声明。

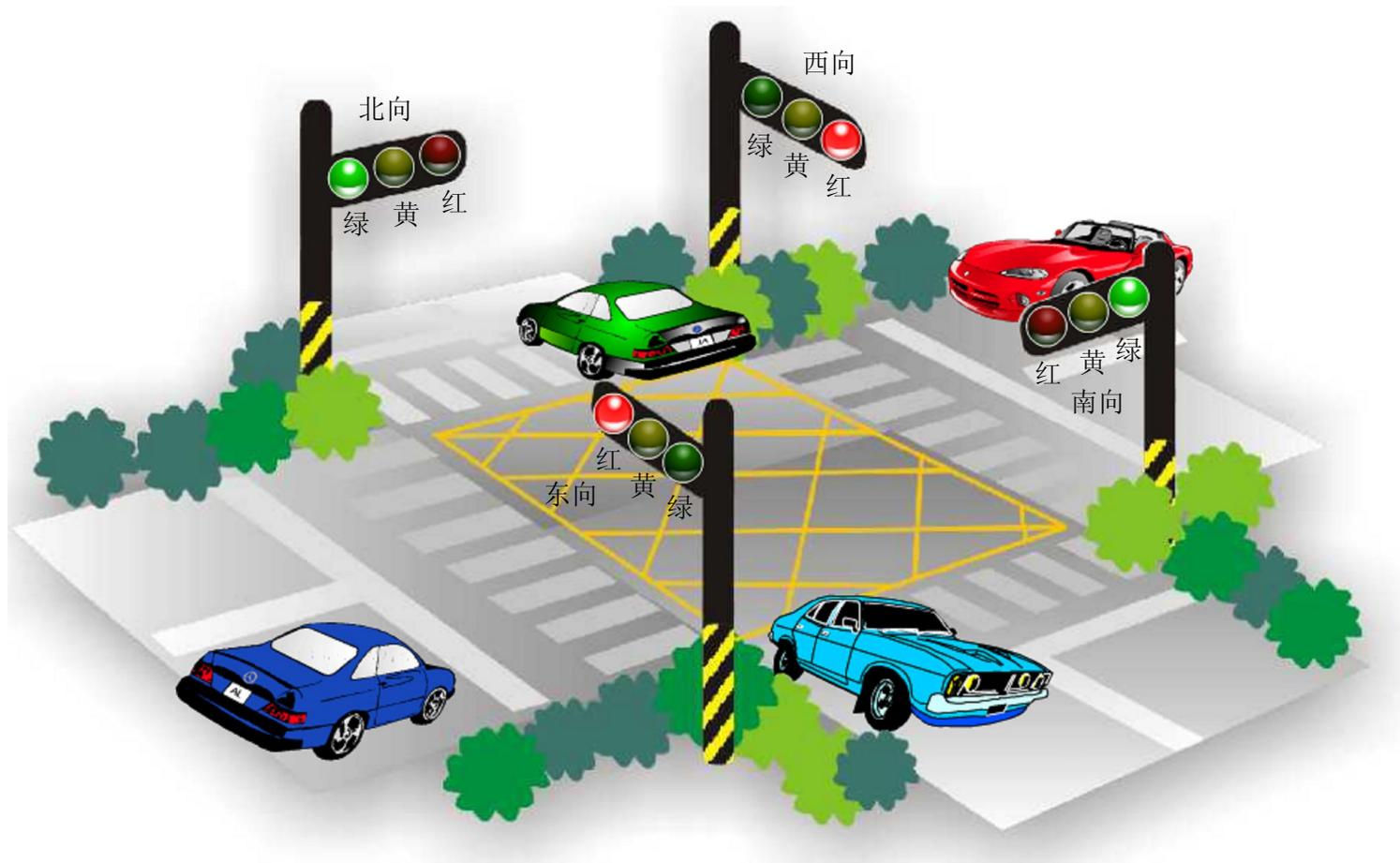
下面以交通信号灯控制系统的设计为例，介绍如何编辑和调用有静态参数的功能块。

#### § 5.7.1 编辑有静态参数的功能块（FB）

#### § 5.7.2 在OB1中调用有静态参数的功能块（FB）

## CH5 S7程序结构与程序设计

### 【例5-7-1】 交通信号灯控制系统程序设计。



返回本节

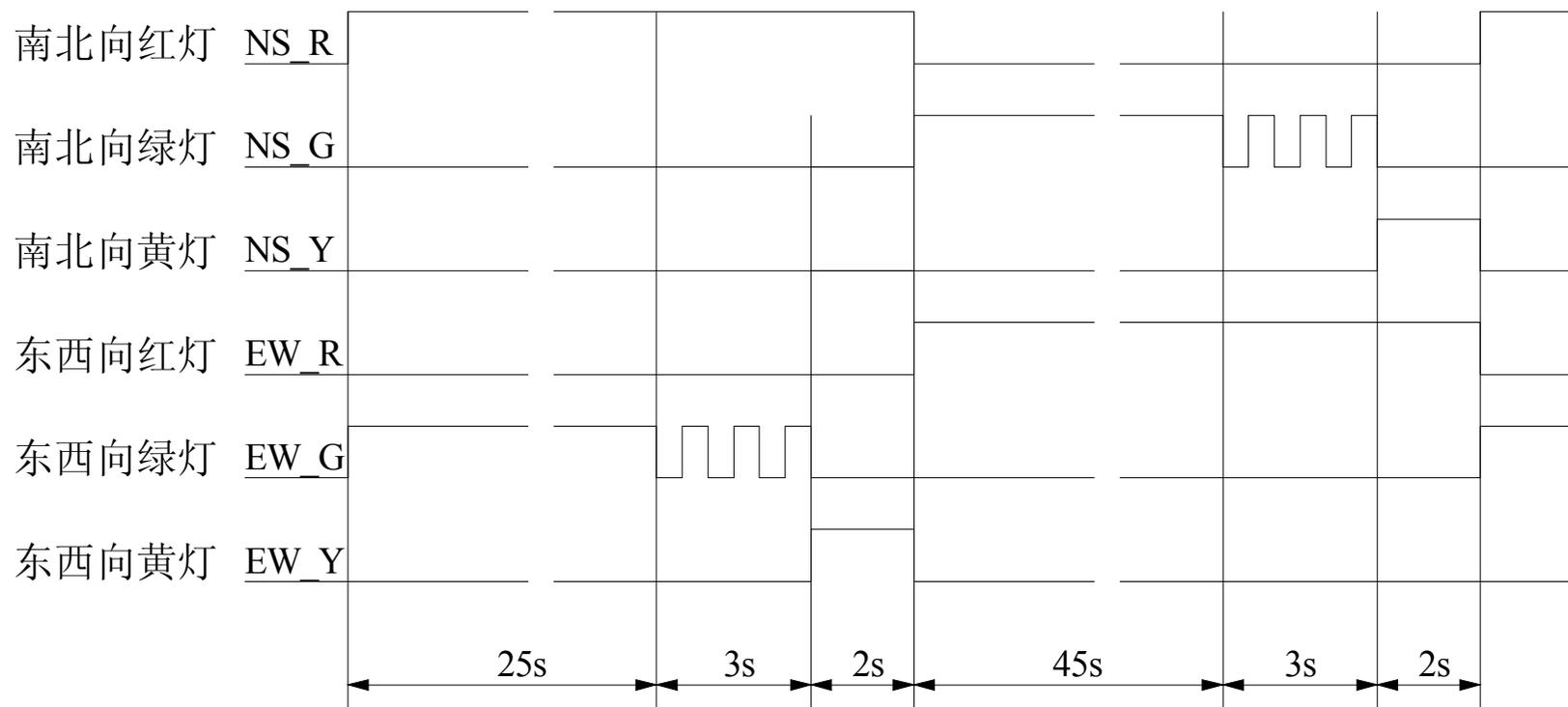
## CH5 S7程序结构与程序设计

上图所示为双干道交通信号灯设置示意图。信号灯的动  
作受开关总体控制，按一下起动按钮，信号灯系统开始工  
作，并周而复始地循环动作；按一下停止按钮，所有信号灯  
都熄灭。信号灯控制的具体要求见下表，试编写信号灯控制  
程序。

南北 方向	信号	SN_G亮	SN_G闪	SN_Y亮	SN_R亮		
	时间	45s	3s	2s	30s		
东西 方向	信号	EW_R亮			EW_G亮	EW_G闪	EW_Y亮
	时间	50s			25s	3s	2s

## CH5 S7程序结构与程序设计

根据十字路口交通信号灯的控制要求，可画出信号灯的  
控制时序图。



返回上级

## CH5 S7程序结构与程序设计

### § 5.7.1 编辑有静态参数的功能块（FB）

- 创建**S7**项目
- 硬件配置
- 编写符号表
- 规划程序结构
- 编辑功能块（**FB**）
- 建立背景数据块（**DI**）
- 编辑启动组织块**OB100**

## CH5 S7程序结构与程序设计

### 1. 创建S7项目

使用菜单【File】→【“New Project”Wizard】创建交通信号灯控制系统的S7项目，并命名为“有静参FB”。项目包  
目  
包  
含组织块OB1和OB100。

### 2. 硬件配置

在“有静参FB”项目内打开“SIMATIC 300 Station”文

Slot	Module	Order number	Fi...	MPI address	I address	Q address	Comment
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU315 (1)	6ES7 315-1AF03-0AB0		2			
3							
4	DI32xDC24V	6ES7 321-1BL80-0AA0			0...3		
5	DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0				4...7	
6							

返回上级

## CH5 S7程序结构与程序设计

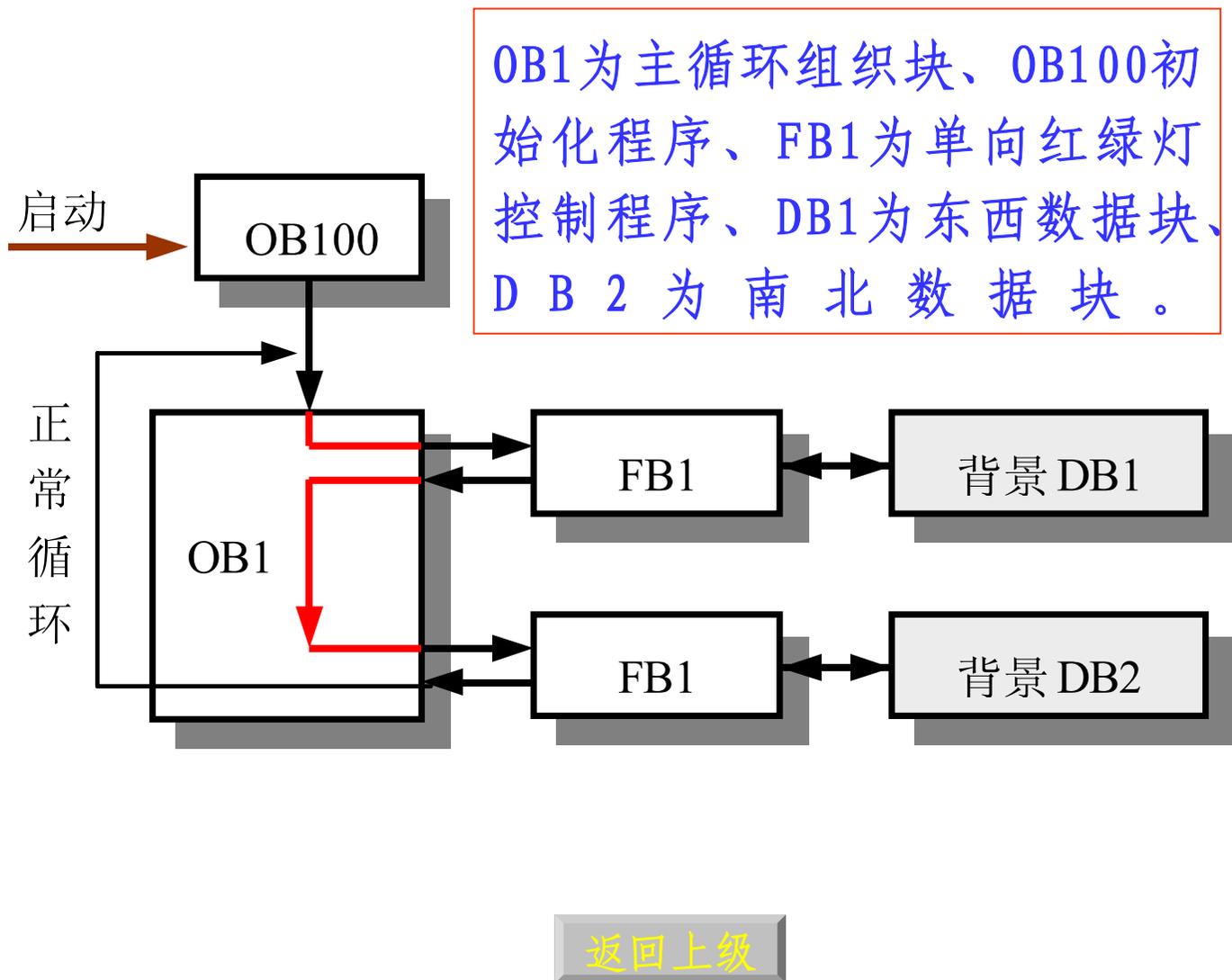
### 3. 编写符号表

	Statu	Symbol ▲	Address	Data typ	Comment
1		Complete Restart	OB 100	OB 100	全启动组织块
2		Cycle Execution	OB 1	OB 1	主循环组织块
3		EW_G	Q 4.1	BOOL	东西向绿色信号灯
4		EW_R	Q 4.0	BOOL	东西向红色信号灯
5		EW_Y	Q 4.2	BOOL	东西向黄色信号灯
6		F_1Hz	M 10.5	BOOL	1Hz时钟信号
7		MB10	MB 10	BYTE	CPU时钟存储器
8		SF	M 0.0	BOOL	系统启动标志
9		SN_G	Q 4.4	BOOL	南北向绿色信号灯
10		SN_R	Q 4.3	BOOL	南北向红色信号灯
11		SN_Y	Q 4.5	BOOL	南北向黄色信号灯
12		Start	I 0.0	BOOL	起动按钮
13		Stop	I 0.1	BOOL	停止按钮
14		T_EW_G	T 1	TIMER	东西向绿灯常亮延时定时器
15		T_EW_GF	T 6	TIMER	东西向绿灯闪亮延时定时器
16		T_EW_R	T 0	TIMER	东西向红灯常亮延时定时器
17		T_EW_Y	T 2	TIMER	东西向黄灯常亮延时定时器
18		T_SN-GF	T 7	TIMER	南北向绿灯闪亮延时定时器
19		T_SN_G	T 4	TIMER	南北向绿灯常亮延时定时器
20		T_SN_R	T 3	TIMER	南北向红灯常亮延时定时器
21		T_SN_Y	T 5	TIMER	南北向黄灯常亮延时定时器
22		东西数据	DB 1	FB 1	为东西向红灯及南北向绿黄灯控制提供实参
23		红绿灯	FB 1	FB 1	红绿灯控制无静态参数的FB
24		南北数据	DB 2	FB 1	为南北向红灯及东西向绿黄灯控制提供实参

返回上级

# CH5 S7程序结构与程序设计

## 4. 规划程序结构



## CH5 S7程序结构与程序设计

### 5. 编辑功能块 (FB)

#### ▶ 定义局部变量声明表

接口类型	变量名	数据类型	地址	初始值	扩展地址	结束地址	注释
In	R_ON	BOOL	0.0	FALSE	-	-	当前方向红灯开始亮标志
	T_R	Timer	2.0	-	-	-	当前方向红色信号灯常亮定时器
	T_G	Timer	4.0	-	-	-	另一方向绿色信号灯常亮定时器
	T_Y	Timer	6.0	-	-	-	另一方向黄色信号灯常亮定时器
In	T_GF	Timer	8.0	-	-	-	另一方向绿色信号灯闪亮定时器
	T_RW	S5Time	10.0	S5T#0MS	-	-	T_R 定时器的初始值
	T_GW	S5Time	12.0	S5T#0MS	-	-	T_G 定时器的初始值
	STOP	BOOL	14.0	S5T#0MS	-	-	停止信号
Out	LED_R	BOOL	10.0	FALSE	-	-	当前方向红色信号灯
	LED_G	BOOL	10.1	FALSE	-	-	另一方向绿色信号灯
	LED_Y	BOOL	10.2	FALSE	-	-	另一方向黄色信号灯
STAT	T_GF_W	S5Time	18.0	S5T#3S	-	-	绿灯闪亮定时器初值
	T_Y_W	S5Time	20.0	S5T#2S	-	-	黄灯常量定时器初值

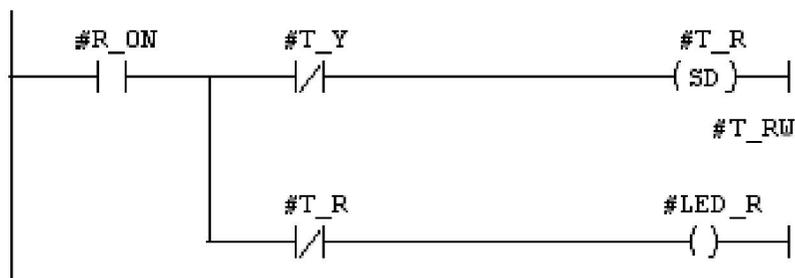
返回上级

# CH5 S7程序结构与程序设计

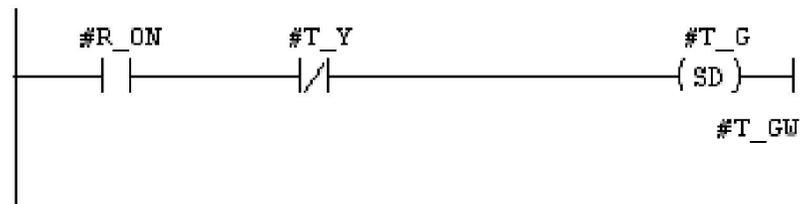
## 编写程序代码

FB1 : 红绿灯控制

**Network 1:** 当前方向红色信号灯延时关闭



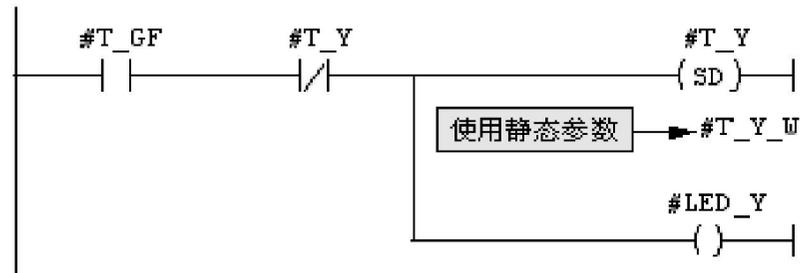
**Network 2:** 另一方向绿色信号灯延时控制



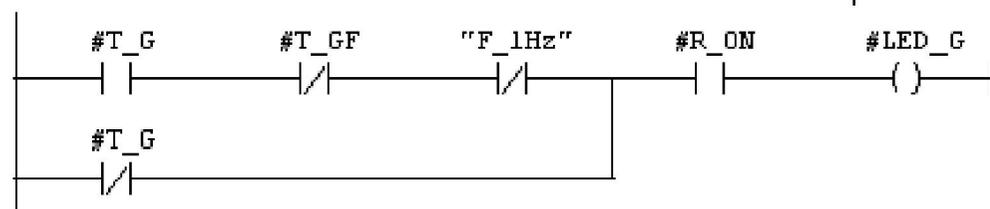
**Network 3:** 启动另一方向绿色信号灯闪亮延时定时器



**Network 4:** 另一方向的黄色信号灯延时控制



**Network 5:** 另一方向的绿色信号灯常亮及闪亮控制

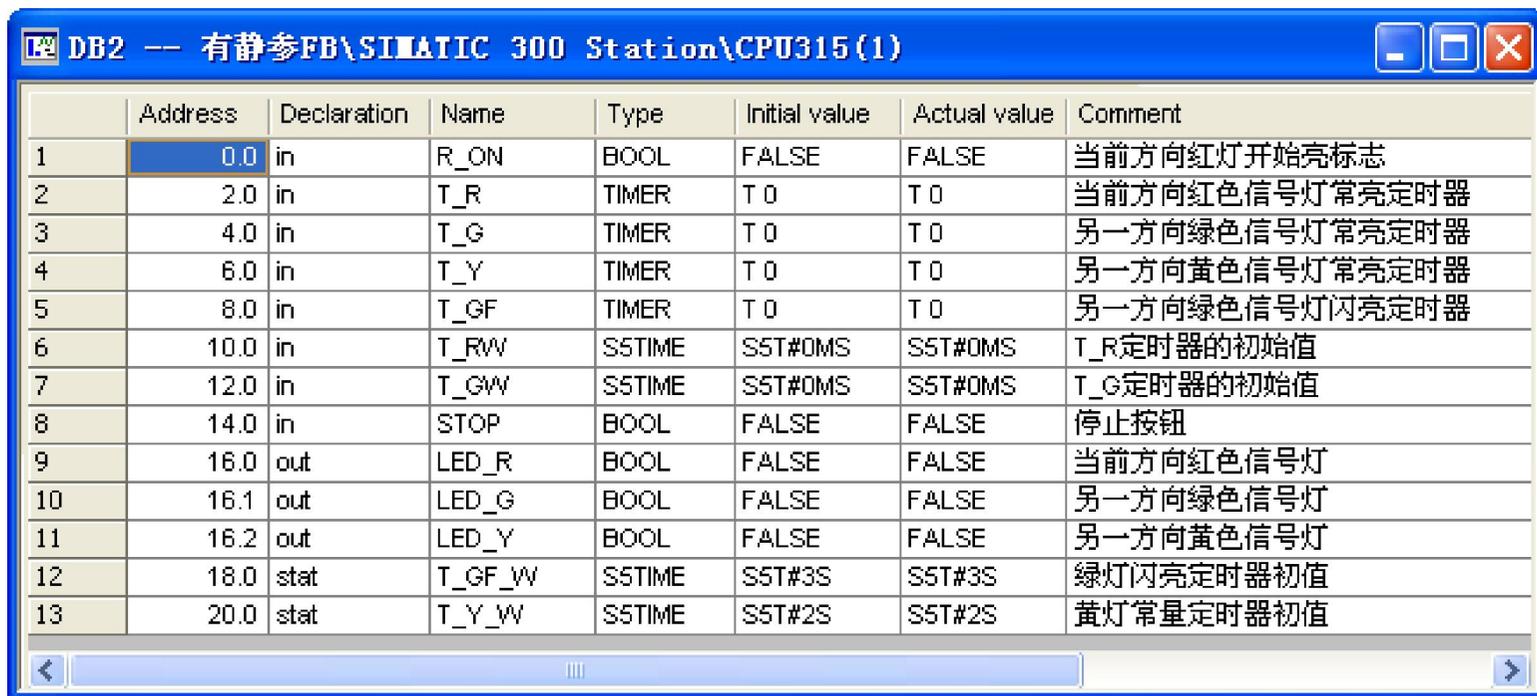


返回上级

## CH5 S7程序结构与程序设计

### 6. 建立背景数据块 (DI)

由于在创建DB1和DB2之前，已经完成了FB1的变量声明，建立了相应的数据结构，所以在创建与FB1相关联的DB1和DB2时，STEP 7自动完成了数据块的数据结构。



The screenshot shows a window titled "DB2 -- 有静参FB\SIMATIC 300 Station\CPU315(1)". It contains a table with the following columns: Address, Declaration, Name, Type, Initial value, Actual value, and Comment. The table lists 13 variables for data block DB2.

	Address	Declaration	Name	Type	Initial value	Actual value	Comment
1	0.0	in	R_ON	BOOL	FALSE	FALSE	当前方向红灯开始亮标志
2	2.0	in	T_R	TIMER	T 0	T 0	当前方向红色信号灯常亮定时器
3	4.0	in	T_G	TIMER	T 0	T 0	另一方向绿色信号灯常亮定时器
4	6.0	in	T_Y	TIMER	T 0	T 0	另一方向黄色信号灯常亮定时器
5	8.0	in	T_GF	TIMER	T 0	T 0	另一方向绿色信号灯闪亮定时器
6	10.0	in	T_RW	S5TIME	S5T#0MS	S5T#0MS	T_R定时器的初始值
7	12.0	in	T_GW	S5TIME	S5T#0MS	S5T#0MS	T_G定时器的初始值
8	14.0	in	STOP	BOOL	FALSE	FALSE	停止按钮
9	16.0	out	LED_R	BOOL	FALSE	FALSE	当前方向红色信号灯
10	16.1	out	LED_G	BOOL	FALSE	FALSE	另一方向绿色信号灯
11	16.2	out	LED_Y	BOOL	FALSE	FALSE	另一方向黄色信号灯
12	18.0	stat	T_GF_WV	S5TIME	S5T#3S	S5T#3S	绿灯闪亮定时器初值
13	20.0	stat	T_Y_WV	S5TIME	S5T#2S	S5T#2S	黄灯常亮定时器初值

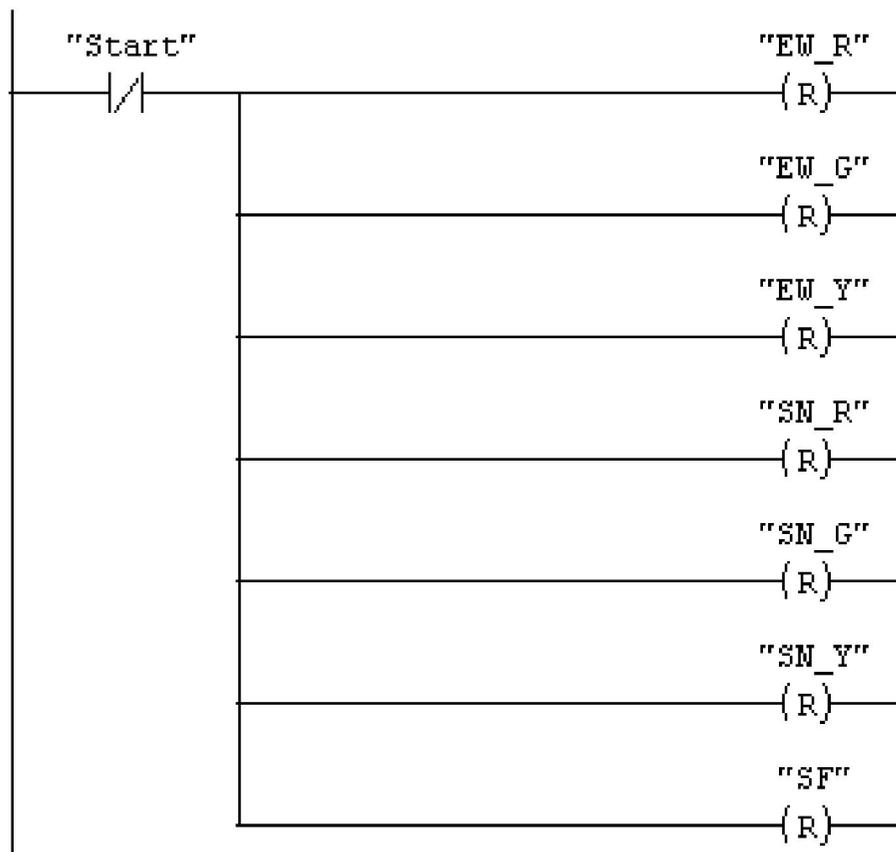
返回上级

# CH5 S7程序结构与程序设计

## 7. 编辑启动组织块OB100

OB100 : "Complete Restart"

**Network 1:** CPU启动时关闭所有信号灯及启动标志



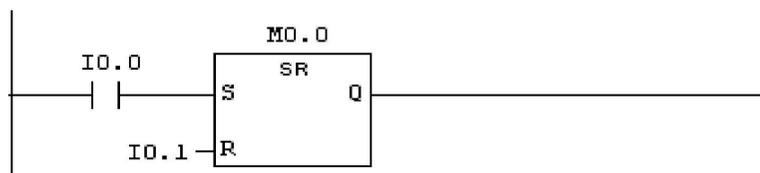
返回上级

# CH5 S7程序结构与程序设计

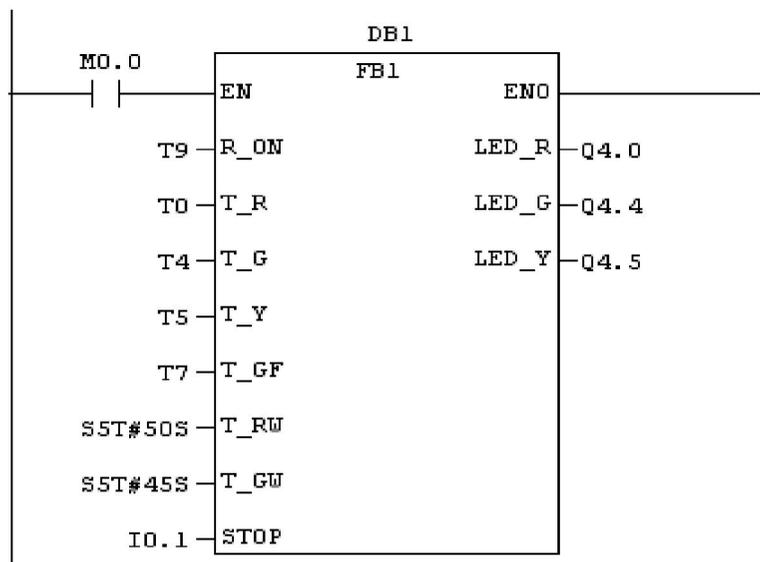
## § 5.7.2 在OB1中调用有静态参数的功能块 (FB)

OB1 : “交通信号灯控制系统的主循环组织块”

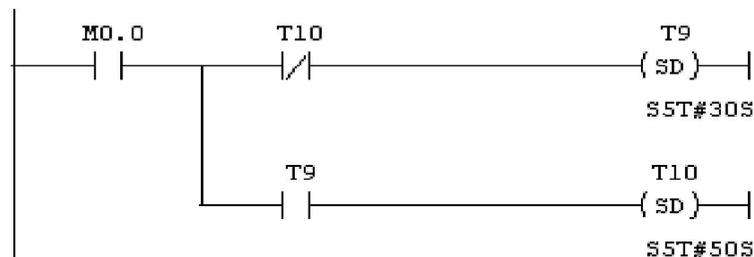
**Network 1:** 设置启动标志



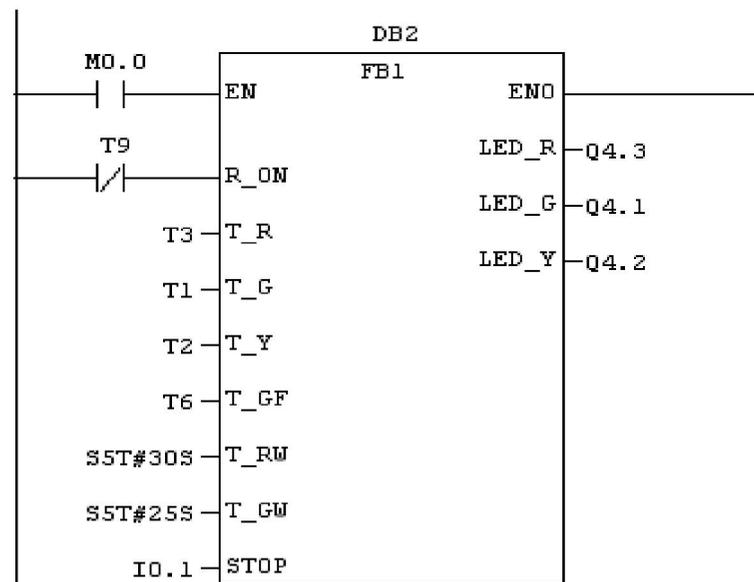
**Network 3:** 东西向红灯及南北向绿灯和黄灯控制



**Network 2:** 设置转换定时器



**Network 4:** 南北向红灯及东西向绿灯和黄灯控制



## § 5.8 使用多重背景——结构化程序设计

使用多重背景可以有效地减少数据块的数量，其编程思想是创建一个比FB1级别更高的功能块，如FB10，将未作任何修改的FB1作为一个“局部背景”，在FB10中调用。对于FB1的每一个调用，都将数据存储在FB10的背景数据块DB10中。

### § 5.8.1 创建多重背景的S7项目

### § 5.8.2 编辑功能（FC）

### § 5.8.3 编辑共享数据块

### § 5.8.4 编辑功能块（FB）

### § 5.8.5 生成多重背景数据块DB10

### § 5.8.6 在OB1中调用功能（FC）及上层功能块（FB）

### § 5.8.1 创建多重背景的S7项目

**【例5-8-1】** 发动机组控制系统设计——使用多重背景。

设某发动机组由1台汽油发动机和1台柴油发动机组成，现要求用PLC控制发动机组，使各台发动机的转速稳定在设定的速度上，并控制散热风扇的起动和延时关闭。每台发动机均设置一个起动按钮和一个停止按钮。

- [创建S7项目](#)
- [硬件配置](#)
- [编写符号表](#)
- [规划程序结构](#)

## CH5 S7程序结构与程序设计

### 1. 创建S7项目

使用菜单【File】→【“New Project”Wizard】创建发电机组控制系统的S7项目，并命名为“多重背景”。CPU选择CPU 315-2DP，项目包含组织块OB1。

### 2. 硬件配置

在“多重背景”项目内打开“SIMATIC 300 Station”文件夹，打开硬件配置窗口，并按下图所示完成硬件配置。

Slot	Module	Order number	Firmware	MPI address	I address	Q address	Comment
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU315-2DP (1)	6ES7 315-2AG10-0AB0	V2.0	2			
2	DP				2047*		
3							
4	DI32xDC24V	6ES7 321-1BL80-0AA0			0...3		
5	DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0				4...7	

返回上级

## CH5 S7程序结构与程序设计

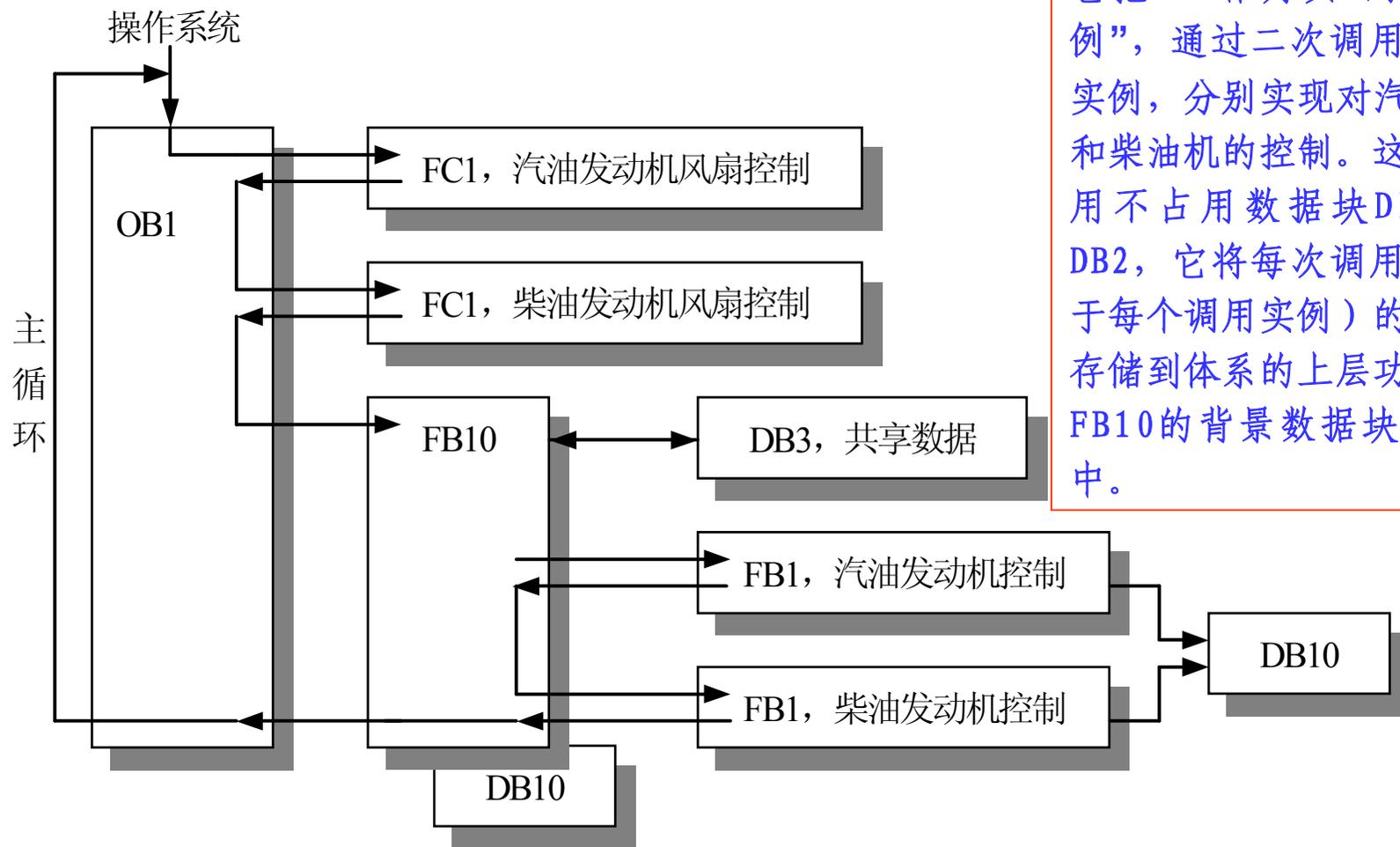
### 3. 编辑符号表

	Statu	Symbol ▲	Address	Data typ	Comment
1		Automatic_Mode	Q 4.2	BOOL	运行模式
2		Automatic_On	I 0.5	BOOL	自动运行模式控制按钮
3		DE_Actual_Speed	MW 4	INT	柴油发动机的实际转速
4		DE_Failure	I 1.6	BOOL	柴油发动机故障
5		DE_Fan_On	Q 5.6	BOOL	启动柴油发动机风扇的命令
6		DE_Follow_On	T 2	TIMER	柴油发动机风扇的继续运行的时间
7		DE_On	Q 5.4	BOOL	柴油发动机的起动机命令
8		DE_Preset_Spe...	Q 5.5	BOOL	显示“已达到柴油发动机的预设转速”
9		Engine	FB 1	FB 1	发动机控制
10		Engine_Data	DB 10	FB 10	FB10的实例数据块
11		Engines	FB 10	FB 10	多重实例的上层功能块
12		Fan	FC 1	FC 1	风扇控制
13		Main_Program	OB 1	OB 1	此块包含用户程序
14		Manual_On	I 0.6	BOOL	手动运行模式控制按钮
15		PE_Actual_Speed	MW 2	INT	汽油发动机的实际转速
16		PE_Failure	I 1.2	BOOL	汽油发动机故障
17		PE_Fan_On	Q 5.2	BOOL	汽油发动机风扇的起动机命令
18		PE_Follow_On	T 1	TIMER	汽油发动机风扇的继续运行的时间
19		PE_On	Q 5.0	BOOL	汽油发动机的起动机命令
20		PE_Preset_Spe...	Q 5.1	BOOL	显示“已达到汽油发动机的预设转速”
21		S_Data	DB 3	DB 3	共享数据块
22		Switch_Off_DE	I 1.5	BOOL	关闭柴油发动机
23		Switch_Off_PE	I 1.1	BOOL	关闭汽油发动机
24		Switch_On_DE	I 1.4	BOOL	起动机柴油发动机
25		Switch_On_PE	I 1.0	BOOL	起动机汽油发动机

返回上级

# CH5 S7程序结构与程序设计

## 4. 规划程序结构



FB10为上层功能块，它把FB1作为其“局部实例”，通过二次调用本地实例，分别实现对汽油机和柴油机的控制。这种调用不占用数据块DB1和DB2，它将每次调用（对于每个调用实例）的数据存储到体系的上层功能块FB10的背景数据块DB10中。

返回上级

### § 5.8.2 编辑功能（FC）

#### 1. 定义局部变量声明表

FC1用来实现发动机（汽油机或柴油机）的风扇控制，按照控制要求，当发动机起动时，风扇应立即起动；当发动机停机后，风扇应延时关闭。因此FC1需要一个发动机起动信号、一个风扇控制信号和一个延时定时器。

接口类型	变量名	数据类型	注释
In	Engine_On	BOOL	发动机的起动信号
In	Timer_Off	Timer	用于关闭延迟的定时器功能
Out	Fan_On	BOOL	起动风扇信号

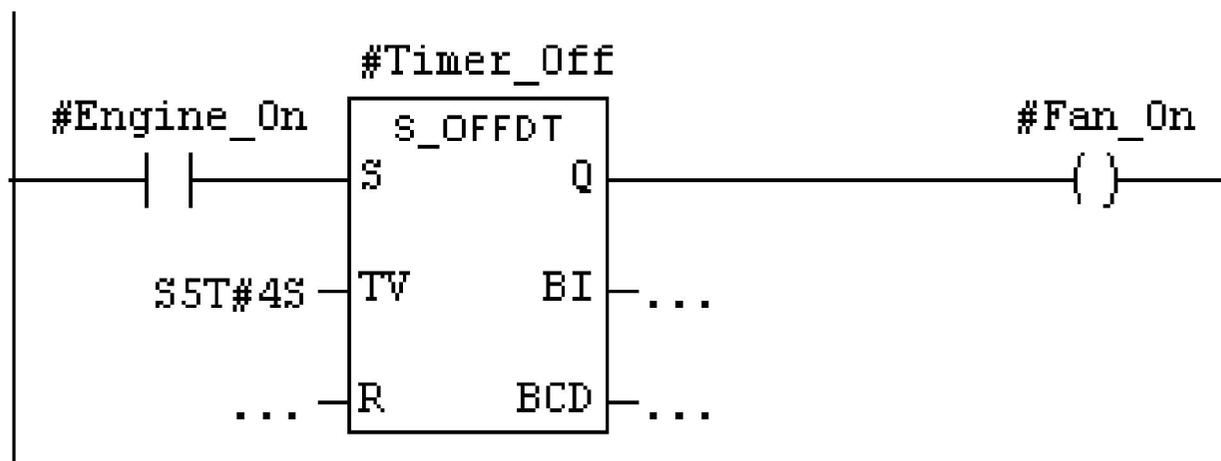
## CH5 S7程序结构与程序设计

### 2. 编辑FC1的控制程序

FC1所实现的控制要求：发动机启动时风扇启动，当发动机再次关闭后，风扇继续运行4s，然后停止。定时器采用断电延时定时器，控制程序如下图所示。

FC1 : 风扇控制功能

**Network 1**: 控制风扇

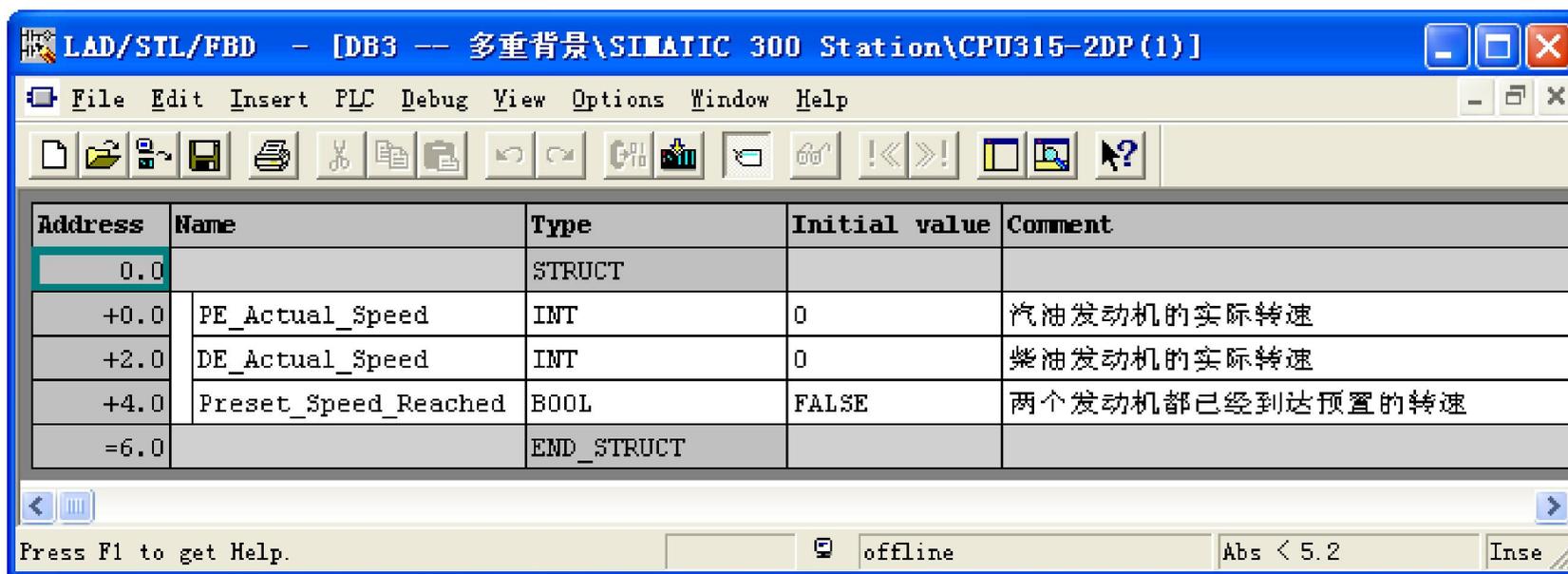


返回上级

## CH5 S7程序结构与程序设计

### § 5.8.3 编辑共享数据块

共享数据块DB3可为FB10保存发动机（汽油机和柴油机）的实际转速，当发动机转速都达到预设速度时，还可以保存该状态的标志数据。



返回本节

### § 5.8.4 编辑功能块（FB）

在该系统的程序结构内，有2个功能块：FB1和FB10。FB1为底层功能块，所以应首先创建并编辑；FB10为上层功能块，可以调用FB1。

➤ 编辑底层功能块FB1

➤ 编辑上层功能块FB10

## CH5 S7程序结构与程序设计

### 1. 编辑底层功能块FB1

在“多重背景”项目内创建FB1，符号名“Engine”。

#### ► 定义功能块**FB1**的变量声明表

接口类型	变量名	数据类型	地址	初始值	扩展地址	结束地址	注释
IN	Switch_On	BOOL	0.0	FALSE	-	-	起动发动机
	Switch_Off	BOOL	0.1	FALSE	-	-	关闭发动机
	Failure	BOOL	0.2	FALSE	-	-	发动机故障，导致发动机关闭
	Actual_Speed	INT	2.0	0	-	-	发动机的实际转速
OUT	Engine_On	BOOL	4.0	FALSE	-	-	发动机已开启
	Preset_Speed_Reached	BOOL	4.1	FALSE	-	-	达到预置的转速
STAT	Preset_Speed	INT	6.0	1500	-	-	要求的发动机转速

返回上级

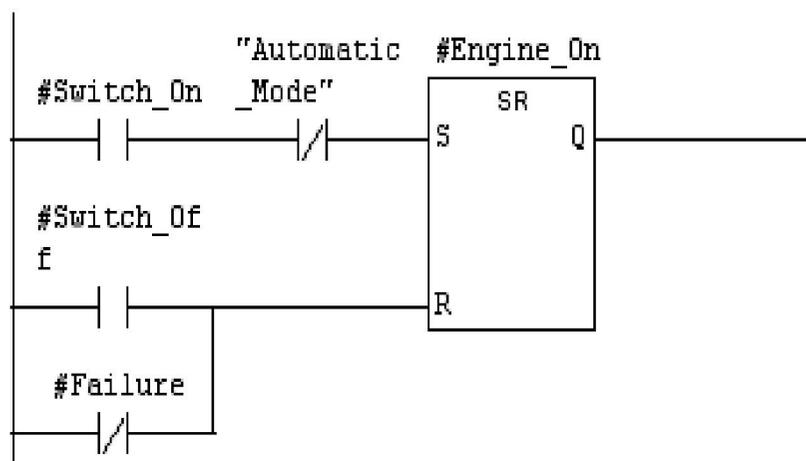
## CH5 S7程序结构与程序设计

### 编写功能块FB1的控制程序

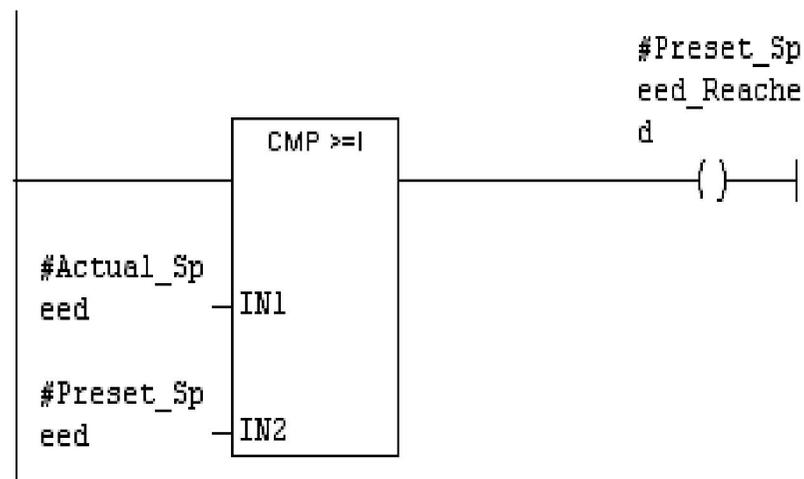
FB1主要实现发动机的起停控制及速度监视功能，其控制程序如下图所示。

FB1 : 发动机控制功能块

**Network 1**: 起动发动机，信号取反



**Network 2**: 监视转速

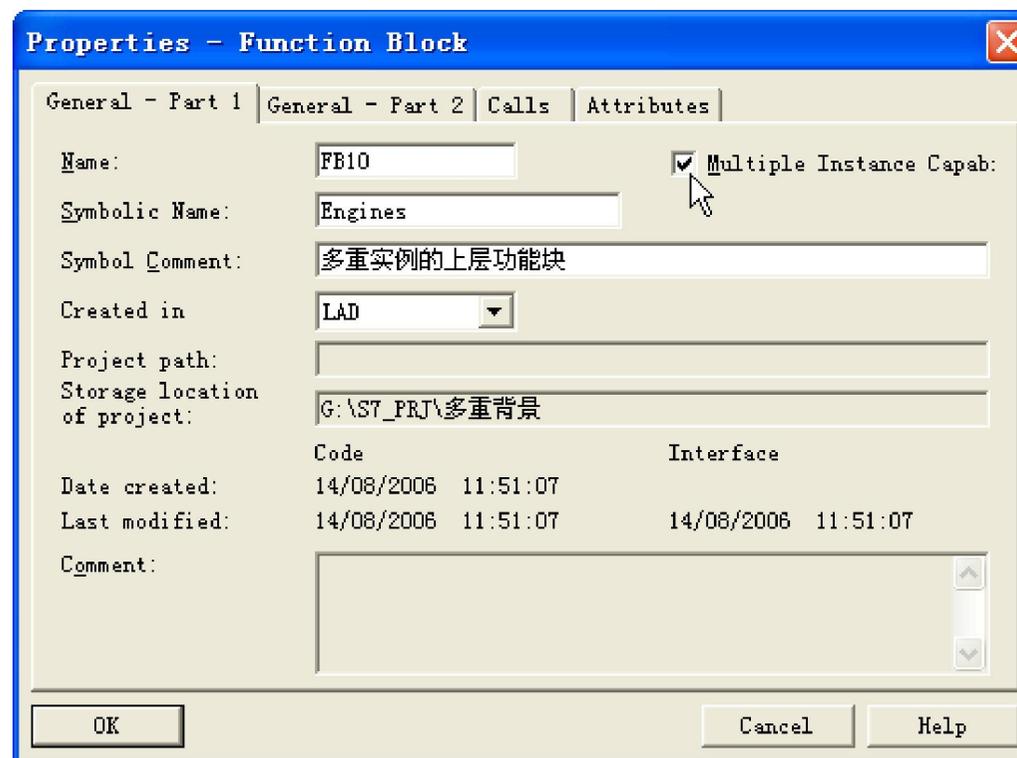


返回上级

## CH5 S7程序结构与程序设计

### 2. 编辑上层功能块FB10

在“多重背景”项目内创建FB10，符号名“Engines”。在FB10的属性对话框内激活“Multi-instance capable”选项。



返回上级

## CH5 S7程序结构与程序设计

### ➤ 定义功能块FB10的变量声明表

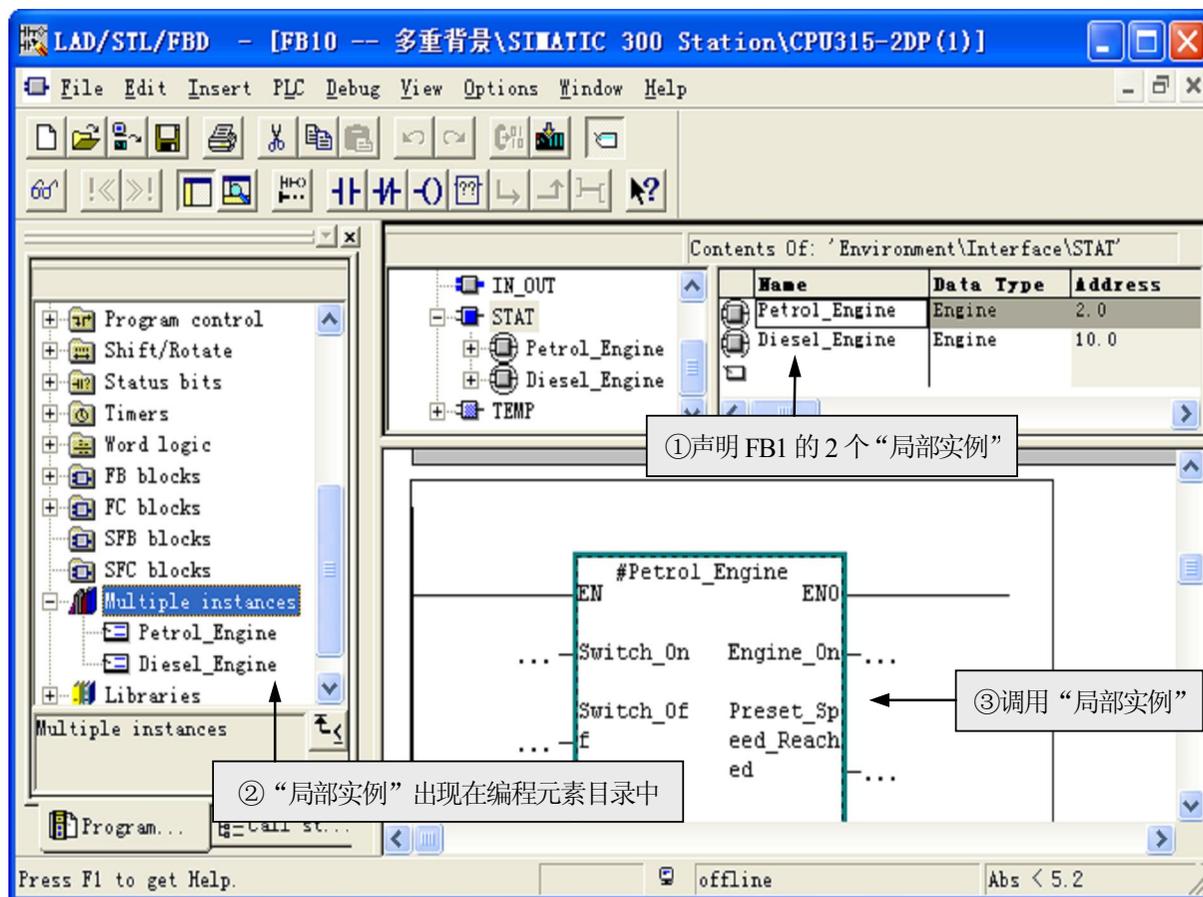
要将FB1作为FB10的一个“局部背景”调用，需要在FB10的变量声明表中为FB1的调用声明不同名称的静态变量，数据类型为FB1（或使用符号名“Engine”）。

接口类型	变量名	数据类型	地址	初始值	注释
OUT	Preset_Speed_Reached	BOOL	0.0	FALSE	两个发动机都已经到达预置的转速。
STAT	Petrol_Engine	FB1	2.0	-	FB1 "Engine" 的第一个局部实例
	Diesel_Engine	FB1	10.0	-	FB1 "Engine" 的第二个局部实例
TEMP	PE_Preset_Speed_Reached	BOOL	0.0	FALSE	达到预置的转速(汽油发动机)
	DE_Preset_Speed_Reached	BOOL	0.1	FALSE	达到预置的转速(柴油发动机)

返回上级

# CH5 S7程序结构与程序设计

## 编写功能块FB10的控制程序



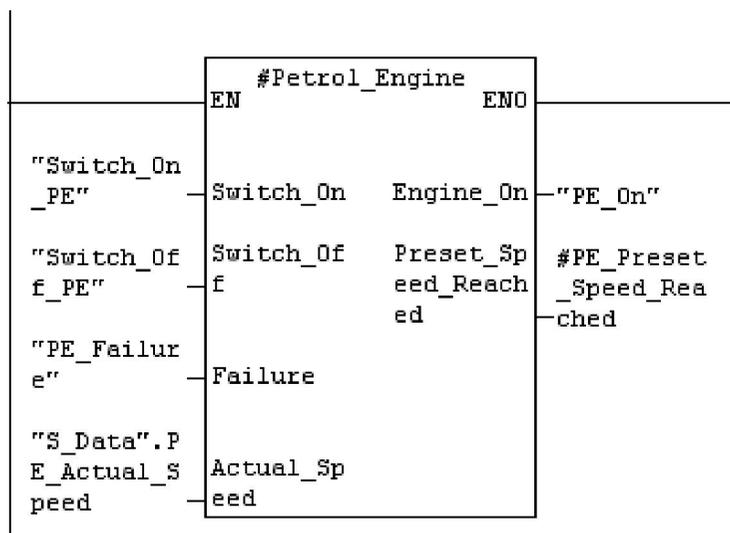
在变量声明表内完成FB1类型的局部实例：“Petrol\_Engine”和“Diesel\_Engine”的声明以后，在程序元素目录的“Multiple Instances”目录中就会出现所声明的多重实例，如图所示。接下来可在FB10的代码区，调用FB1的“局部实例”。

# CH5 S7程序结构与程序设计

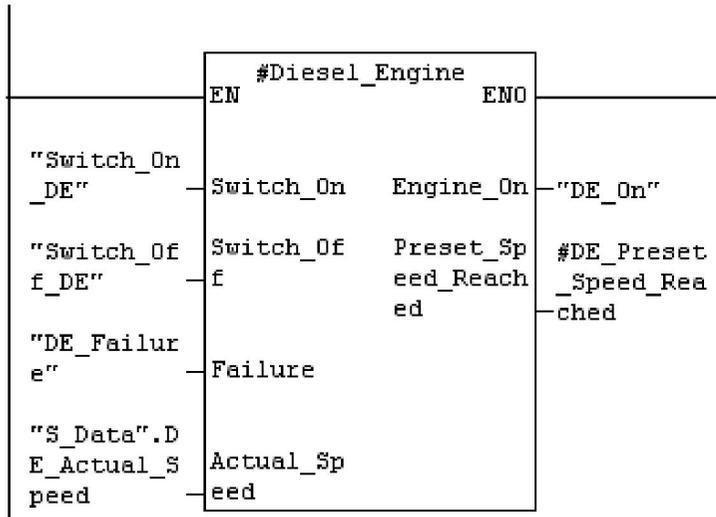
## 编写功能块FB10的控制程序

FB10 : 多重背景

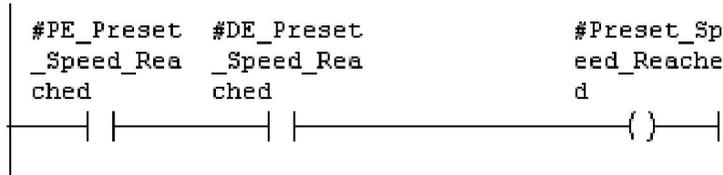
**Network 1:** 启动汽油发动机



**Network 2:** 启动柴油发动机



**Network 3:** 两台发动机均已达到设定转速

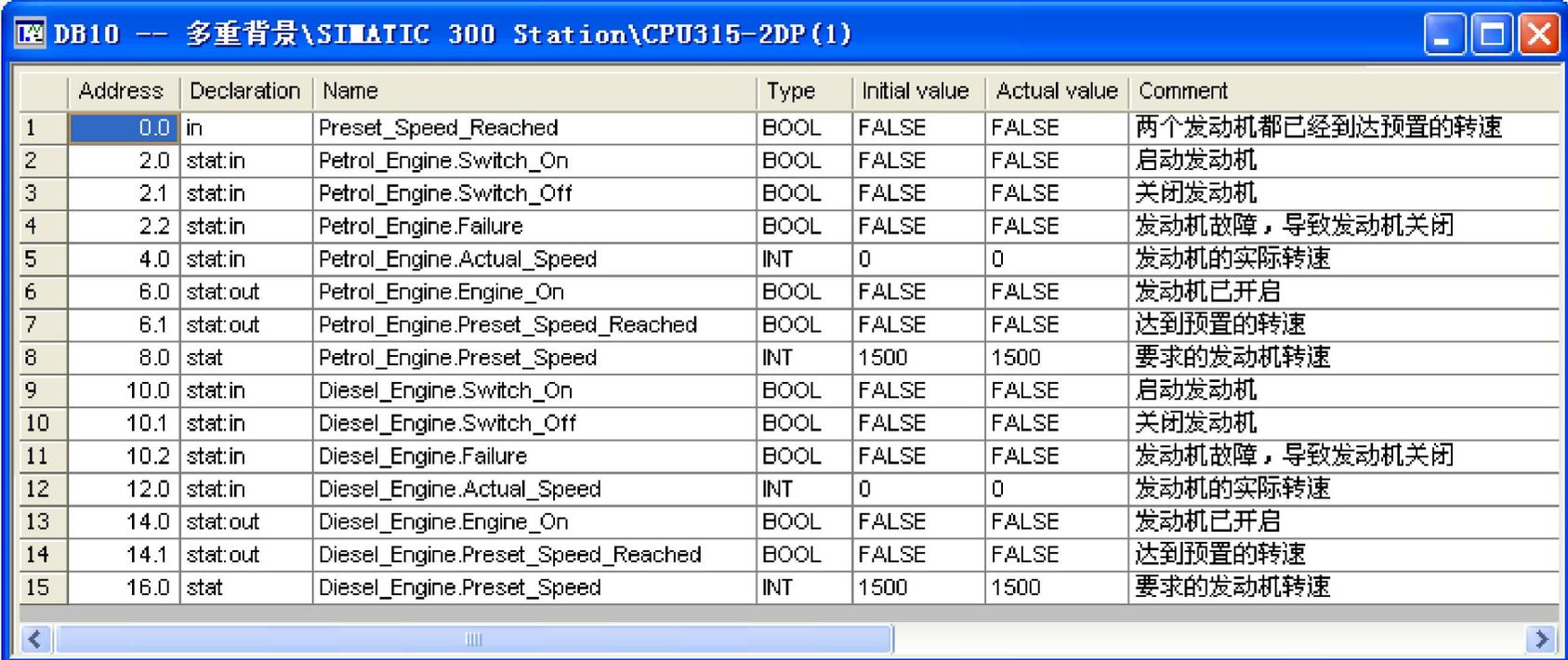


调用FB1局部实例时，不再使用独立的背景数据块，FB1的实例数据位于FB10的实例数据块DB10中。发动机的实际转速可直接从共享数据块中得到，如DB3.DBW2（符号地址为，S\_Data".PE\_Actual\_Speed）。

## CH5 S7程序结构与程序设计

### § 5.8.5 生成多重背景数据块DB10

在“多重背景”项目内创建一个与FB10相关联的多重背景数据块DB10，符号名“Engine\_Data”。



	Address	Declaration	Name	Type	Initial value	Actual value	Comment
1	0.0	in	Preset_Speed_Reached	BOOL	FALSE	FALSE	两个发动机都已经到达预置的转速
2	2.0	stat:in	Petrol_Engine.Switch_On	BOOL	FALSE	FALSE	启动发动机
3	2.1	stat:in	Petrol_Engine.Switch_Off	BOOL	FALSE	FALSE	关闭发动机
4	2.2	stat:in	Petrol_Engine.Failure	BOOL	FALSE	FALSE	发动机故障，导致发动机关闭
5	4.0	stat:in	Petrol_Engine.Actual_Speed	INT	0	0	发动机的实际转速
6	6.0	stat:out	Petrol_Engine.Engine_On	BOOL	FALSE	FALSE	发动机已开启
7	6.1	stat:out	Petrol_Engine.Preset_Speed_Reached	BOOL	FALSE	FALSE	达到预置的转速
8	8.0	stat	Petrol_Engine.Preset_Speed	INT	1500	1500	要求的发动机转速
9	10.0	stat:in	Diesel_Engine.Switch_On	BOOL	FALSE	FALSE	启动发动机
10	10.1	stat:in	Diesel_Engine.Switch_Off	BOOL	FALSE	FALSE	关闭发动机
11	10.2	stat:in	Diesel_Engine.Failure	BOOL	FALSE	FALSE	发动机故障，导致发动机关闭
12	12.0	stat:in	Diesel_Engine.Actual_Speed	INT	0	0	发动机的实际转速
13	14.0	stat:out	Diesel_Engine.Engine_On	BOOL	FALSE	FALSE	发动机已开启
14	14.1	stat:out	Diesel_Engine.Preset_Speed_Reached	BOOL	FALSE	FALSE	达到预置的转速
15	16.0	stat	Diesel_Engine.Preset_Speed	INT	1500	1500	要求的发动机转速

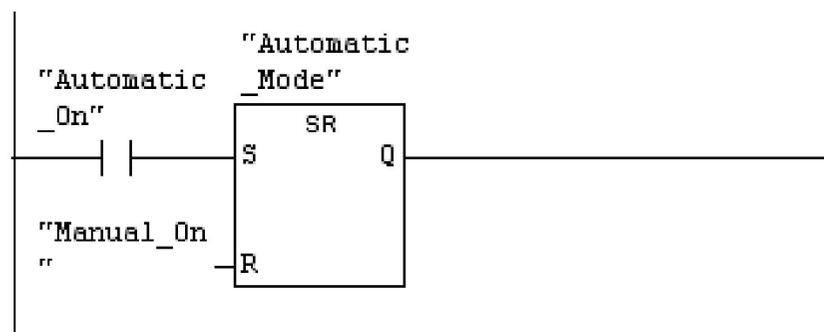
返回本节

# CH5 S7程序结构与程序设计

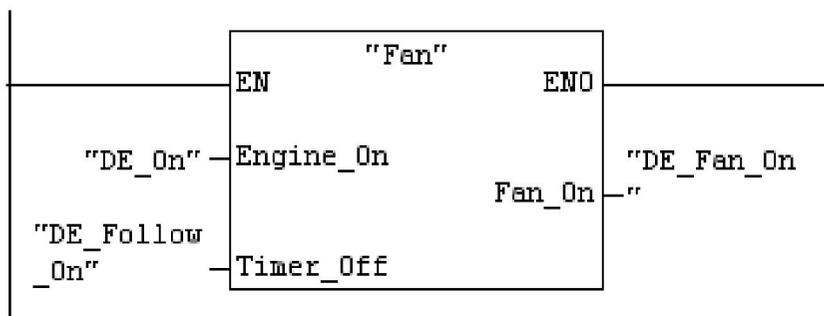
## § 5.8.6 在OB1中调用功能(FC)及上层功能块(FB)

OB1 : 主循环程序

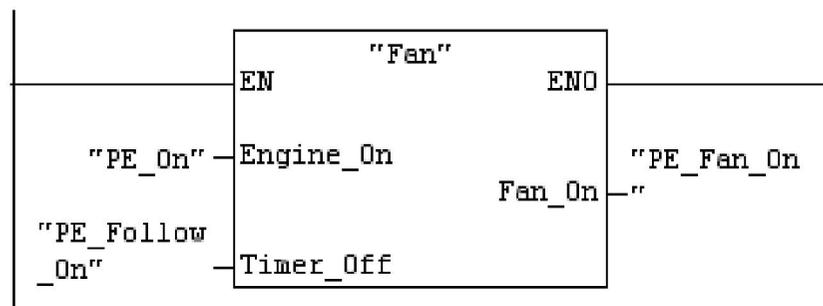
**Network 1:** 设置运行模式



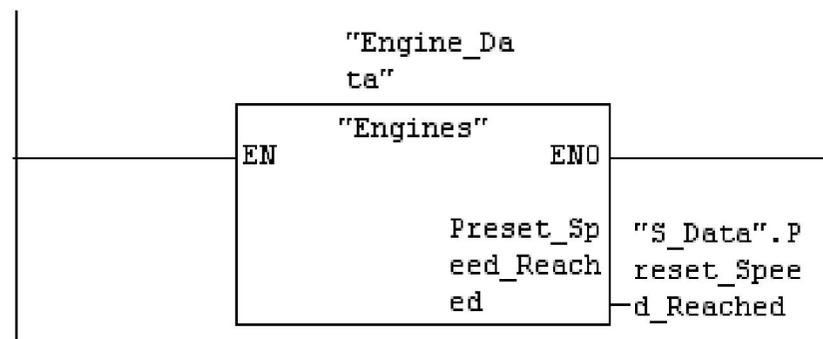
**Network 3:** 控制柴油发动机风扇



**Network 2:** 控制汽油发动机风扇



**Network 4:** 调用上层功能块FB10



## § 5.9 思考与练习

### 一、思考题

1. STEP 7中有哪些逻辑块？
2. 功能FC和功能块FB有何区别？
3. 系统功能SFC和系统功能块有何区别？
4. 共享数据块和背景数据块有何区别？
5. 什么是符号地址？采用符号地址有哪些好处？
6. 组织块可否调用其他组织块？
7. B堆栈与L堆栈有何不同？
8. 在变量声明表内，所声明的静态变量和临时变量有何区别？

## CH5 S7程序结构与程序设计

### 二、编程题

1. 用I0.0控制接在Q4.0~Q4.7上的8个彩灯循环移位，用T37定时，每0.5 s移1位，首次扫描时给Q4.0~Q4.7置初值，用I0.1控制彩灯移位的方向，试设计语句表程序。

2. 有一工业用洗衣机，控制要求如下：

①按起动按钮后给水阀就开始给水→当水满传感器动作时就停止给水→波轮正转5s，再反转5s，然后再正转5s如此反复转动5分钟→出水阀开始出水→出水10s后停止出水，同时声光报警器报警，叫工作人员来取衣服。

②按停止按钮声光报警器停止，并结束工作过程。

**要求：**分配I/O口，设计梯形图。

## CH5 S7程序结构与程序设计

### 3. 车辆出入库管理。

如图5-76所示为车辆入库管理设备布置图，编制一个用PLC控制的车辆出入库管理梯形图控制程序，控制要求如下：

① 入库车辆前进时，经过1#传感器→2#传感器后计数器加1，后退时经过2#传感器→1#传感器后计数器减1，单经过一个传感器则计数器不动。

② 出库车辆前进时经过2#传感器→1#传感器后计数器减1，后退时经过1#传感器→2#传感器后计数器加1，单经过一个传感器则计数器不动作。

③ 设计一个由两位数码管及相应的辅助元件组成的显示电路，显示车库内车辆的实际数量。

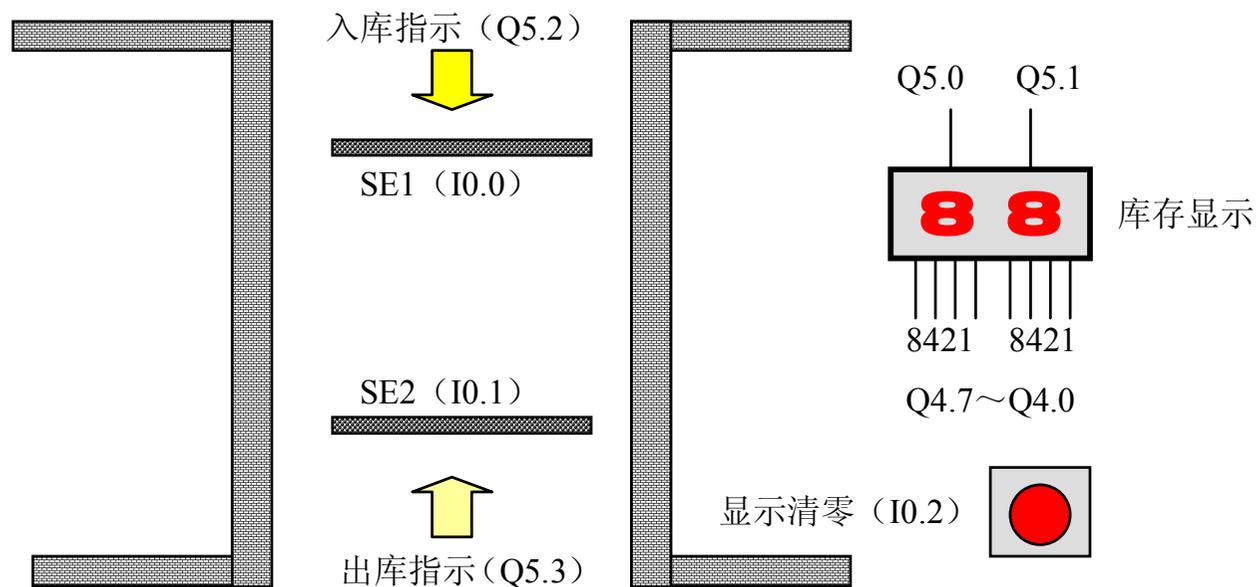


图5-76 车辆入库管理

## CH5 S7程序结构与程序设计

4. 液体自动混合箱如图5-77所示，设计要求如下：

按下起动按钮SB1，电磁阀K1打开，液体A流入箱中，当液面到达L2处时，K1阀关闭，同时K2阀打开，液体B流入箱中，当液面到达L1处时，Y2阀关闭，停止供液，电炉H开始加热，当液体到达指定温度时，温度传感器T动作，电炉停止加热，搅拌机M开始搅拌液体，5分钟后停止搅拌，K3阀打开，将加热并混合好的液体放出，当液面底于L3时，再经过10s，K3阀关

闭，此时箱内液体已放空。此时，电磁阀K1打开，液体A流入箱中，开始下一周期循环。按下停止按钮SB2，系统停止操作（停在初始状态上）。

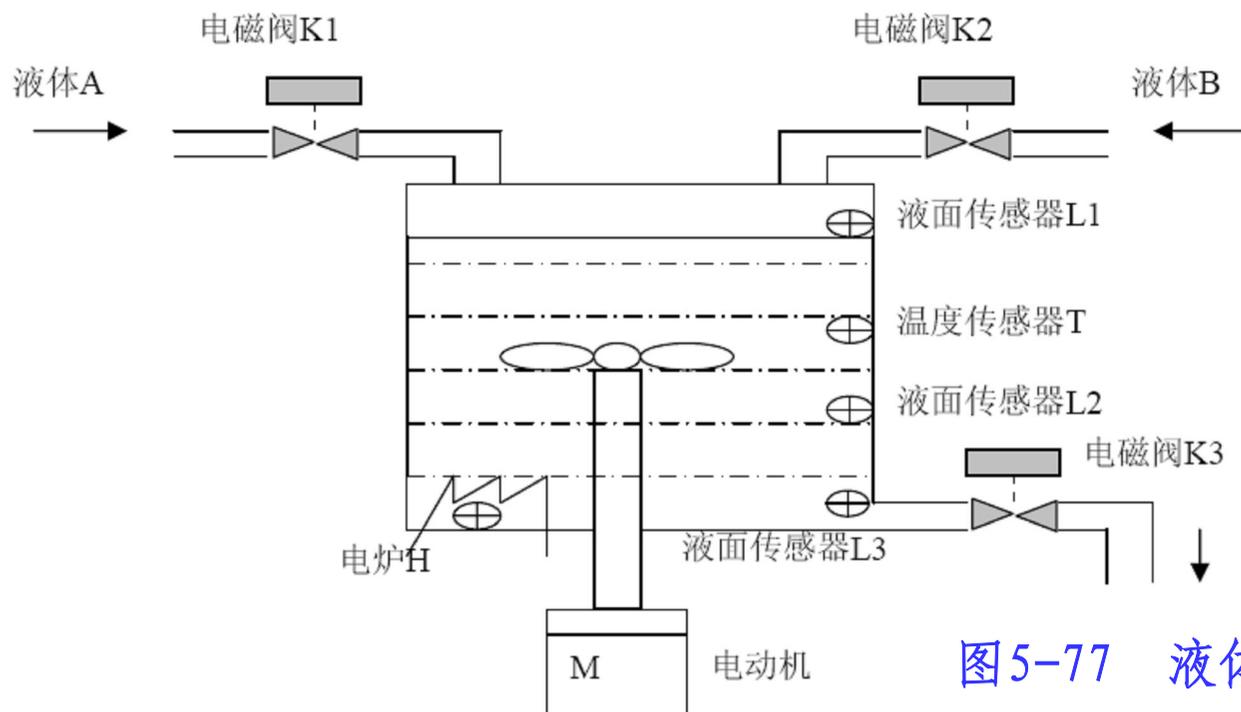


图5-77 液体自动混合箱

## CH5 S7程序结构与程序设计

5. 如图5-78所示为精密滚柱直径筛选系统，当被测滚柱落下后，由气缸推杆推到限位挡板位置，然后钨钢测头开始测试滚柱直径，并将测量值送相敏检波放大器处理，再送电压放大器放大，最后将与直径成正比的电压值送PLC模拟量输入模块，经PLC判断后，根据直径大小来决定具体打开哪一个翻版，然后由电磁机构将限位挡板抽离，滚柱自然落入相应的容器中。试设计PLC控制程序，实现对滚柱的筛选。

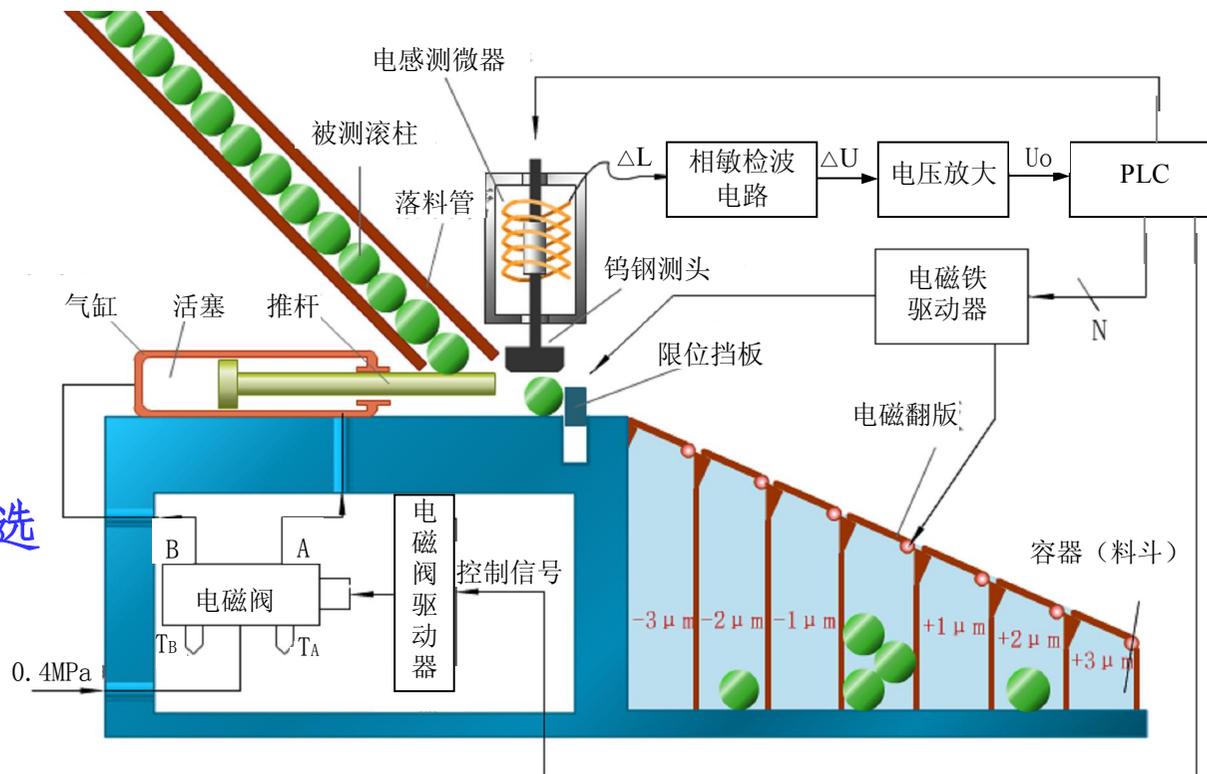


图5-78 滚柱直径筛选系统

## CH5 S7程序结构与程序设计

5. 有一工业用洗衣机，控制要求如下：

①按起动按钮后给水阀就开始给水→当水满传感器动作时就停止给水→波轮正转5s，再反转5s，然后再正转5s如此反复转动5分钟→出水阀开始出水→出水10s后停止出水，同时声光报警器报警，叫工作人员来取衣服。

②按停止按钮声光报警器停止，并结束整个工作过程。

要求：分配I/O口，设计梯形图。

6. 使用传送机，将大、小球分类后分别传送的系统。

左上为原点，按起动按钮SB1后，其动作顺序为：下降→吸球（延时1s）上升→右行→下降→放球（延时1s）→上升→左行。

其中：LS1左限位；LS3上限位；LS4小球右限位；LS5大球右限位；LS2大球下限位；LS0小球下限位。

机械臂下降时，吸住大球，则下限位LS2接通，然后将大球放到大球容器中。若吸住小球，则下限位LS0接通，然后将小球放到小球容器中。

试分配I/O，设计画梯形图。