

# 组织块使用说明

## 1、组织块总览 (OB)

### 1. 1 什么是组织块?

组织块是操作系统和用户程序之间的接口。OB 用于执行具体的程序:

- 在CPU 启动时
- 在一个循环或时钟执行时
- 当发生故障时
- 当发生硬件中断时

组织块根据其优先级执行。

### 1. 2 可使用哪些组织块?

不是所有的CPU 都能处理STEP 7 中所有的组织块。至于您使用的CPU 中包括哪些OB, 请参见操作列表/72/和/102/。

何处可以找到更多信息?

参见在线帮助和以下手册可以得到更多的信息:

- /70/: 该手册包含有描述不同S7-300 CPU 功能的技术数据。
- /101/: 该手册包含有描述不同S7-400 CPU 功能的技术数据。下列表格中包含每一个OB 的启动事件及对应的优先级。

OB	启动事件	默认的优先级	解释
OB1	启动结束或OB1 执行结束	1	自由循环
OB10	日期时间中断0	2	没有指定缺省时间
OB11	日期时间中断1	2	
OB12	日期时间中断2	2	
OB13	日期时间中断3	2	
OB14	日期时间中断4	2	
OB15	日期时间中断5	2	
OB16	日期时间中断6	2	
OB17	日期时间中断7	2	
OB20	延时中断0	3	没有指定缺省时间
OB21	延时中断1	4	
OB22	延时中断2	5	
OB23	延时中断3	6	

OB	启动事件	默认的优先级	解释
OB30	循环中断0 (缺省时间间隔: 5s)	7	循环中断
OB31	循环中断1 (缺省时间间隔: 2s)	8	
OB32	循环中断2 (默认时间间隔: 1s)	9	

OB33	循环中断3 (默认时间间隔: 500ms)	10	
OB34	循环中断4 (默认时间间隔: 200ms)	11	
OB35	循环中断5 (默认时间间隔: 100ms)	12	
OB36	循环中断6 (默认时间间隔: 50ms)	13	
OB37	循环中断7 (默认时间间隔: 20ms)	14	
OB38	循环中断8 (默认时间间隔: 10ms)	15	
OB40	硬件中断0	16	硬件中断
OB41	硬件中断1	17	
OB42	硬件中断2	18	
OB43	硬件中断3	19	
OB44	硬件中断4	20	
OB45	硬件中断5	21	
OB46	硬件中断6	22	
OB47	硬件中断7	23	
OB55	状态中断	2	DPV1 中断
OB56	刷新中断	2	
OB57	制造厂商用特殊中断	2	
OB60	SFC 35 “MP_ALM” 调用	25	多处理器中断
OB61	周期同步中断1	25	同步循环中断
OB62	周期同步中断2	25	
OB63	周期同步中断3	25	
OB64	周期同步中断4	25	
OB70	I/O 冗余故障 (只对于H CPU)	25	冗余故障中断
OB72	CPU 冗余故障 (只对于H CPU)	28	
OB73	通讯冗余故障 (只对于H CPU)	25	
OB80	时间故障	26, 281)	同步故障中断
OB81	电源故障	25, 281)	
OB82	诊断中断	25, 281)	
OB83	模板插/拔中断	25, 281)	
OB84	CPU 硬件故障	25, 281)	
OB85	程序故障	25, 281)	
OB86	扩展机架、DP 主站系统或分布式 I/O 从站故障	25, 281)	
OB87	通讯故障	25, 281)	
OB88	过程中断	28	
OB90	暖或冷启动或删除一个正在OB90 中执行的块或装载一个OB90 到CPU 或中止OB90	292)	背景循环

OB	启动事件	默认的优先级	解释
OB100	暖启动	271)	启动
OB101	热启动	271)	
OB102	冷启动	271)	
OB121	编程故障	引起故障的OB的 优先级 引起故障的OB的 优先级	同步故障中断
OB122	I/O 访问故障		

<sup>1)</sup> 优先级27 和28 在优先级启动模式中是有效的。<sup>2)</sup> 优先级29 对应于优先级 0. 29。这意味着背景循环比自由循环具有更低的优先级。

## 2. 组织块说明

### 2. 1 程序循环组织块 (OB1)

#### 2. 1. 1 描述

S7 CPU 操作系统周期性地执行组织块OB1 程序。当OB1 执行完毕，操作系统再次启动它。CPU 启动后，OB1 循环执行。您可以在OB1 中调用其它功能块 (FB, SFB ) 或功能 (FC, SFC ) 。

#### 2. 1. 2 理解OB1 的运行

OB1 的优先级最低。其循环时间被监控。即除OB90 以外，其它所有OB 均可中断OB1 的执行。以下事件可导致操作系统调用OB1：

- CPU 启动完毕。
- OB1 执行到上一个循环周期结束。

OB1 执行完后，操作系统发送全局数据。再次启动OB1 之前，操作系统会将输出映像区数据写入输出模板，刷新输入映像区并接收全局数据。S7 监视最长循环时间，保证最长的响应时间。最长循环时间缺省设置为150ms。您可以设一个新值或通过SFC43 “RE\_TRIGR” 重新启动时间监视功能。如果您的程序超过了OB1 最长循环时间，操作系统将调用OB80 (时间故障OB)；如果OB80 不存在，则CPU 停机。

除了监视最长循环时间，还可以保证最短循环时间。操作系统将延长下一个新循环 (将输

出映像区数据传送到输出模板)直到最短循环时间到。参见手册/70/和/101/参数“最长”、“最短”循环时间的范围。您可以运用STEP 7 软件更改参数设置。

#### 2. 1. 3 OB1 的局部数据

以下表格描述了OB1 的临时变量 (TEMP)。变量名是OB1 的缺省名称。

变量	类型	描述
OB1_EV_CLASS	BYTE	事件等级和标识码: B#16#11: OB1 激活

OB1_SCAN_1	BYTE	• B#16#01 : 暖启动完成• B#16#02 : 热启动完成• B#16#03 : 主循环完成• B#16#04 : 冷启动完成• B#16#05: 当前一个主站CPU 停机, 后备新主站CPU 的第一次OB1 循环
OB1_PRIORITY	BYTE	优先级1
OB1_OB_NUMBR	BYTE	OB 号 (01)
OB1_RESERVED_1	BYTE	备用
OB1_RESERVED_2	BYTE	备用
OB1_PREV_CYCLE	INT	上一次OB1 的循环时间 (ms)
OB1_MIN_CYCLE	INT	自CPU 启动, 最短一次OB1 的循环时间 (ms)
OB1_MAX_CYCLE	INT	自CPU 启动, 最长一次OB1 的循环时间 (ms)
OB1_DATE_TIME	DATE_AND_TIME	OB 被调用的日期和时间

## 2. 2 日期时间中断组织块 (OB10 到OB17)

### 2. 2. 1 描述

STEP 7 提供多达8 个OB (OB10 到OB17) , 它们可以运行一次或周期性地运行。您可以运用SFC 或STEP 7 给CPU 分配参数, 这样OB 将以下列的间隔运行:

- 一次
- 每分钟
- 每小时
- 每天
- 每周
- 每月
- 每月底

### 2. 2. 2 理解日期时间中断组织块OB 的运行

在启动日期时间中断时, 您必须首先设置和激活中断。

以下三种方式可以设置和激活中断:

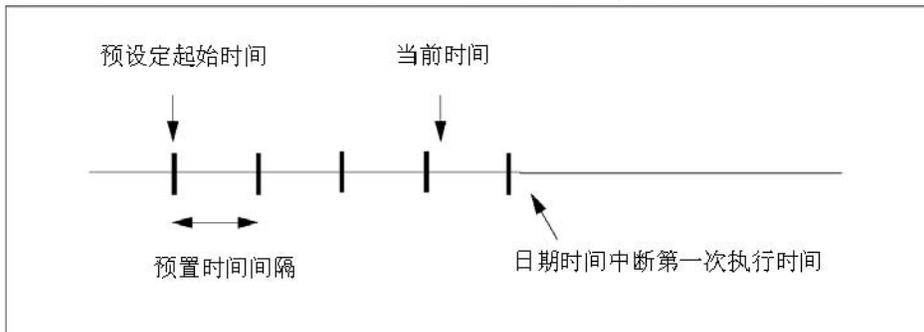
- 自动启动日期时间中断。您可通过STEP 7 设置并激活中断。下表显示通过STEP 7 激活日期时间中断的基本可能性。
- 您可以在STEP 7 中设置日期时间中断, 然后通过程序调用SFC30 “ACT-TINT” , 激活日期时间中断。
- 您可以通过调用SFC28 “ SET\_TINT ” 设置日期时间中断, 通过调用SFC30 “ACT\_TINT” , 激活日期时间中断。

时间间隔	描述
没有激活	日期时间中断没有执行, 即使已装载到CPU。可以通过调用

	SFC30 激活。
只激活一次	日期时间中断运行一次后，日期时间中断OB 自动取消。您可以通过调用SFC28 和SFC30 重新设置和重新激活日期时间中断OB。
周期性地激活	当日期时间中断发生后，CPU 根据当前的时间计算下一次的日期时间中断启动时间。

如果您设定日期时间中断相应的OB 是执行一次，那么日期时间（DATE\_AND\_TIME ）不能是过去（与CPU 的实时时钟相关）的日期时间。如果您设定日期时间中断相应的OB 是周期性的执行，日期时间（DATE\_AND\_TIME ）是过去的日期和时间，那么日期时间中断将按下图所示在下次执行。

您可以用SFC39 到SFC42 禁止、延迟和重新使能日期时间中断。



### 2. 2. 3 影响日期时间中断OB 的条件

因为日期时间中断只在规定的的时间间隔发生，某些条件会在程序执行时影响OB 的运行。下面表格显示了一些影响日期时间中断OB 运行的条件和描述。

条件	结果
您编程调用SFC29（CAN_TINT），取消了日期时间中断。	操作系统清除日期时间中断的事件日期时间（DATE_AND_TIME）。在此OB 被调用之前，您必须重新设置并激活它。
您编程激活日期时间中断OB，但此OB 在CPU 中不存在。	操作系统调用OB85。如果OB85 在CPU 中不存在，CPU 将停机。

条件	结果
当同步或校正CPU 系统时钟时，您设置的时间超前并跳过了日期时间中断OB 的启动事件。	操作系统调用OB80 并在OB80 中编译日期时间中断OB 号和启动事件信息。操作系统运行一次日期时间中断OB，不考虑这个OB 应该运行多少次。OB80 的启动事件信息显示了OB 第一次被跳过的日期和时间（DATE_AND_TIME）。

当同步或校正CPU 系统时钟时，时间被向后设置，于是OB 的启动事件、日期或时间被重复。	如果日期时间OB 在时钟修改之前已激活，那么此OB 不再被调用。
CPU 在暖启动或冷启动期间。	任何由SFC 配置的日期时间OB， 都须遵从STEP7 中组态的设置。如果您组态了一个启动一次的日期时间中断OB 在STEP7 中，组态的时间是过去（相对于CPU 的实时时钟）， 并且已激活，在操作系统暖启动或冷启动后，这个 OB 将被调用一次。
下一个时间间隔启动事件发生时，日期时间OB 仍在执行。	操作系统调用OB80。如果OB80 不存在，则CPU 停机。如果OB80 已装载，那么OB80 和日期时间中断OB 都执行第一次，并且第二个中断请求也被执行。

## 2. 2. 4 日期时间中断OB 的局部数据

下列表格描述了日期时间中断的临时变量（TEMP）。变量名为OB10 的缺省名称。

变量	类型	描述
OB10_EV_CLASS	BYTE	事件级和识别码：B#16#11=中断激活
OB10_STRT_INFO	BYTE	B#16#11：启动请求OB10（B#16#12：启动请求OB11）：：（B#16#18：启动请求OB17）
OB10_PRIORITY	BYTE	分配的优先级：默认2
OB10_OB_NUMBR	BYTE	OB 号（10 到17）
OB10_RESERVED_1	BYTE	保留
OB10_RESERVED_2	BYTE	保留
OB10_PERIOD_EXE	WORD	OB 以特殊的间隔运行：W#16#0000：一次W#16#0201：每分钟一次W#16#0401：每小时一次W#16#1001：每天一次W#16#1201：每周一次W#16#1401：每月一次W#16#1801：每年一次W#16#2001：每月底
OB10_RESERVED_3	INT	保留
OB10_RESERVED_4	INT	保留
OB10_DATE_TIME	DATE_AND_TIME	OB 调用时的日期和时间

## 2. 3 延时中断组织块（OB20 到OB23）

### 2. 3. 1 描述

S7 提供多达4 个OB（OB20 到OB23），它们经过一段指定的延时后运行。每一个OB

在调用SFC32（SRT\_DINT）后启动。延时时间在SFC 的参数中设定。当您编程SFC32（SRT\_DINT）时，需提供OB 号、延时时间和一个用户指定的标识符。指定的延时到，OB 启动。您也可以取消还没有启动的延时中断。

### 2. 3. 2 理解延时中断OB 的运行

当延时时间到（毫秒时间和OB 号一起传给SFC32）时，操作系统启动相应的OB。运用延时中断，您必须做下列事项：

- 您必须调用SFC32（SRT\_DINT）；
- 您必须下载日期时间中断OB 到CPU 中。

延时OB 只有在CPU 处于运行状态时才运行。一个暖启动或冷启动清除任何延时OB 的启动事件。如果延时中断还没有启动，您可以调用SFC33（CAN\_DINT）去取消它的执行。延时时间分辨率为1 毫秒，如果延时时间超出则立即再次启动延时。通过调用 SFC34

（QRY\_DINT）可以查询延时中断的状态。

如果以下事件发生，操作系统调用一个异步OB：

- 如果您在调用SFC32（SRT\_DINT）时启动的是没有下载到CPU 中的OB。
- 如果一个延时中断OB 执行未结束，下一个延时中断启动事件发生。您可以运用SFC 39 到SFC 42 取消、延时和再使能延时中断。

### 2. 3. 3 延时中断组织块OB 的局部数据

下面表格描述了延时中断OB 的临时变量。变量名是OB20 的缺省名称。

变量	类型	描述
OB20_EV_CLASS	BYTE	事件级别和识别码： B#16#11： 中断激活
OB20_STRT_INF	BYTE	B#16#21： OB20 启动请求（B#16#21： OB21 启动请求）（B#16#22： OB22 启动请求）（B#16#23： OB23 启动请求）
OB20_PRIORITY	BYTE	分配的优先级： 默认值为3（OB20）到6（OB23）
OB20_OB_NUMBR	BYTE	OB 号（20 到23）
OB20_RESERVED_1	BYTE	保留
OB20_RESERVED_2	BYTE	保留
OB20_SIGN	WORD	用户ID： SFC32（SRT_DINT）的输入参数SIGN

变量	类型	描述
OB20_DTIME	TIME	以毫秒形式组态的延时时间
OB20_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

## 2. 4 循环中断组织块 (OB30 到OB38)

### 2. 4. 1 描述

S7 提供多达9 个循环中断组织块OB (OB30 到OB38)，它们经过一段固定的时间间隔中断您的程序。下面表格显示了循环中断OB 默认的时间间隔和优先级。

OB 号	默认的时间间隔	默认的优先级
OB30	5 秒	7
OB31	2 秒	8
OB32	1 秒	9
OB33	500 毫秒	10
OB34	200 毫秒	11
OB35	100 毫秒	12
OB36	50 毫秒	13
OB37	20 毫秒	14
OB38	10 毫秒	15

### 2. 4. 2 理解循环中断组织块OB 的运行

循环中断OB 的等距时间间隔，是由时间间隔和相位偏移量确定。参见/234/ 中OB 启动时间、时间周期和相位偏移量的关系。

注意：您必须确认循环中断OB 每一次运行的时间一定要短于中断的间隔。如果一个循环中断OB 没有执行完，循环中断时间到，又要求循环中断OB 运行，则时间故障组织块OB80 启动。循环中断导致故障程序的运行。

您可以运用SFC39 到42 取消或延时和再次使能循环中断。对于您使用的CPU 参数时间间隔、优先级、相位偏移量的设置范围请参见相应CPU 的描述。您可以用STEP7 软件修改参数设置。

### 2. 4. 3 循环中断组织块OB 的局部数据

下面表格描述了循环中断组织块OB 的临时变量 (TEMP)。变量名称是OB35 的缺省名称。

变量	类型	描述
OB35_EV_CLASS	BYTE	事件级别和识别码 B#16#11：中断激活
OB35_STRT_INF	BYTE	B#16#30： 循环中断组织块OB 的启动请求，只对于特殊标准（只有H 型CPU 并且明确地为其组态）B#16#31：OB30 启动请求： B#16#36：OB35 启动请求： B#16#39：OB38 启动请求
OB35_PRIORITY	BYTE	分配的优先级：默认7（OB30）到15（OB38）
OB35_OB_NUMBR	BYTE	OB 号（30 到38）
OB35_RESERVED_1	BYTE	保留

OB35_RESERVED_2	BYTE	保留
OB35_PHASE_OFFSET	WORD	相位偏移量[毫秒]
OB35_RESERVED_3	INT	保留
OB35_EXC_FREQ	INT	时间间隔，以毫秒计
OB35_DATE_TIME	DATE_AND_TIME	OB 调用时的日期和时间

## 2. 5 硬件中断组织块 (OB40 到OB47)

### 2. 5. 1 描述

S7 提供多达8 个独立的硬件中断组织块OB (OB40 到OB47) 。通过STEP 7 进行参数赋值，可以为能够触发硬件中断的每一个信号模板指定以下参数。

- 哪个通道在哪种条件下触发一个硬件中断。
- 一个硬件中断OB 被分配到单独的通道组。(作为默认,所有硬件中断被OB40 处理)

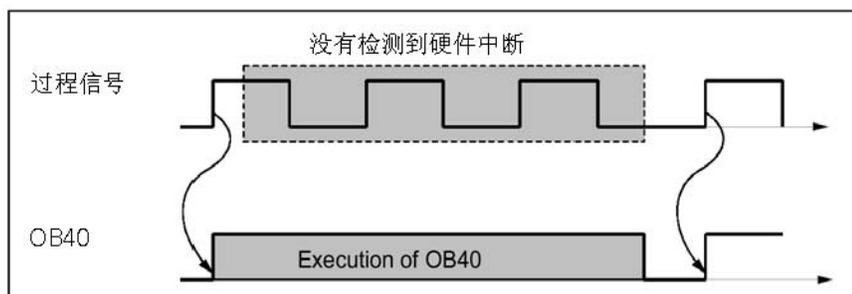
运用CP 和FM 模板，您可以用它们自己的软件设置这些参数。

您可以运用STEP 7 为每一个硬件中断OB 选择优先级。

### 2. 5. 2 理解硬件中断组织块OB 的运行

在硬件中断被模板触发之后，操作系统识别相应的槽和相应的硬件中断OB。如果这个OB比当前激活的OB 优先级高，则启动该OB。在硬件中断OB 执行之后，将发送通道确认。如果在处理硬件中断的同时，同一中断模板上有另一个硬件中断，这个新的中断的识别与确认过程如下：

- 如果事件发生在以前触发硬件中断的通道，旧的硬件中断触发程序正在执行，则新中断丢失。如下面图所示。图中例子是一个数字量输入模板的通道。触发信号是上升沿。硬件中断OB 是OB40。
- 如果这个事件发生在同一模板的另一个通道，那么没有硬件中断能被触发。但是这个



中断没有丢失，在确认当前激活硬件之后被触发。如果一个硬件中断触发并且它的OB 正在由于另一个模板的硬件中断而激活着，则记录新的中断申请，在空闲后会执行该中断。

用SFC 39 - 42 可以禁止、延时和再使能硬件中断。您可以用STEP 7 中的SFC 55 到57，为硬件中断模板分配参数。

### 2. 5. 3 硬件中断OB 的局部数据

下列表格中描述了硬件中断OB 的临时变量 (TEMP) 。变量名是OB40 的缺省名称。

变量	类型	描述
----	----	----

OB40_EV_CLASS	BYTE	事件级别和诊断号： B#16#11： 中断被激活
OB40_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>• B#16#41 : 中断通过中断行1</li> <li>• B#16#42 : 中断通过中断行2 (只对S7-400)</li> <li>• B#16#43 : 中断通过中断行3 (只对S7-400)</li> <li>• B#16#44 : 中断通过中断行4(只对S7-400)</li> <li>• B#16#45: WinAC 通过PC 触发的中断</li> </ul>
OB40_PRIORITY	BYTE	分配优先级:默认16(OB40)到23(OB47)
OB40_OB_NUMBR	BYTE	OB 号 (40 到47)
OB40_RESERVED_1	BYTE	保留
OB40_IO_FLAG	BYTE	输入模板: B#16#54 输出模板: B#16#55
OB40_MDL_ADDR	WORD	触发中断模块的逻辑地址
OB40_POINT_ADDR	DWORD	<ul style="list-style-type: none"> <li>• 数字模板: 带有模板输入状态的位字段 (0 位对应第一个输入)</li> <li>• 模拟模板: 带有限幅信息输入通道的位字段 (结构详见/71/ 或/101/)</li> <li>• CP 或IM: 模块中断状态 (不是与用户相关的)</li> </ul>
OB40_DATE_TIME	DATE_AND_TIME	OB 被调用的日期和时间

注意: 如果您用有DPV1 能力的CPU, 您可以通过SFB54 “RELRM” 获取额外的信息, 此信息超出了OB 的启动信息。当您在S7 兼容模式下运行工业现场总线DP 主站时, 此信息也会提供。

## 2. 6 状态中断组织块 (OB 55)

注意: 一个状态中断组织块OB (OB 55) 只在有DPV1 能力的CPU 中存在。

### 2. 6. 1 描述

如果状态中断通过DPV1 从站槽触发, CPU 操作系统调用OB55。这可能是一个DPV1 从站组件更改其操作模式的情况。例如从运行到停止。事件触发一个状态中断的详细信息, 参见DPV1 从站手册。

### 2. 6. 2 状态中断组织块OB 的局部变量

下面表格包含了状态中断的临时变量 (TEMP)。变量名是OB55 的缺省名称。

变量	类型	描述
OB55_EV_CLASS	BYTE	事件级别和标识: B#16#11 (即将发生的事件)
OB55_STRT_INF	BYTE	B#16#55: (OB55 启动请求)
OB55_PRIORITY	BYTE	组态的优先级, 默认值为2
OB55_OB_NUMBR	BYTE	OB 号 (55)
OB55_RESERVED_1	BYTE	备用
OB55_IO_FLAG	BYTE	输入模板: B#16#54 输出模板: B#16#55
OB55_MDL_ADDR	WORD	中断触发组件 (模板) 的逻辑地址
OB55_LEN	BYTE	中断提供的数据块长度

OB55_TYPE	BYTE	“状态中断”的中断ID号
OB55_SLOT	BYTE	中断触发组件（模板）的槽号
OB55_SPEC	BYTE	详细说明•位0至1：中断详细说明•位2：ADD_Ack • 位3至7：Seq. No.e
OB55_DATE_TIME	DATE_AND_TIME	OB 被调用的日期时间

注意：您可以在OB55中通过调用SFB54“RALRM”，从工业现场总线DP信息报文中可以得到更新的信息。

## 2.7 刷新中断组织块（OB 56）

注意：一个刷新中断组织块OB（OB56）只有在有DPV1能力的CPU中存在。

### 2.7.1 描述

如果刷新中断通过DPV1从站槽触发，CPU运行系统调用OB56。这可能是如果您变换了DPV1从站槽的参数（通过本地或远程访问）。详细的刷新中断触发事件信息，参见DPV1从站手册。

### 2.7.2 刷新中断组织块OB的局部变量

下面表格包含了更新中断的临时变量（TEMP）。变量名是OB56的缺省名称。

变量	类型	描述
OB56_EV_CLASS	BYTE	事件级别和标识：B#16#11（即将发生的事件）
OB56_STRT_INF	BYTE	B#16#56（OB56启动请求）
OB56_PRIORITY	BYTE	组态的优先级，默认值为2
OB56_OB_NUMBR	BYTE	OB号（56）
OB56_RESERVED_1	BYTE	备用
OB56_IO_FLAG	BYTE	输入模板：B#16#54 输出模板：B#16#55
OB56_MDL_ADDR	WORD	中断触发组件（模板）的逻辑地址
OB56_LEN	BYTE	中断提供的数据块长度
OB56_TYPE	BYTE	“刷新中断”的中断ID号
OB56_SLOT	BYTE	中断触发组件（模板）的槽号
OB56_SPEC	BYTE	详细说明•位0至1：中断详细说明•位2：ADD_Ack • 位3至7：Seq. No.
OB56_DATE_TIME	DATE_AND_TIME	OB调用的日期时间

注意：您可以在OB55中通过调用SFB54“RALRM”，从工业现场总线D信息报文中可以得到更多信息。

## 2.8 制造商用中断组织块（OB57）

注意：制造商指定的中断组织块OB（OB57）只有在有DPV1能力的CPU中存在。

### 2.8.1 描述

如果制造商用中断通过DPV1从站槽触发，CPU运行系统调用OB57。

### 2.8.2 制造商用中断组织块OB的局部变量

下表包含了制造商用中断的临时变量（TEMP）。变量名是OB57的缺省名称。

变量	类型	描述
OB57_EV_CLASS	BYTE	事件级别和标识：B#16#11（即将发生

		的事件)
OB57_STRT_INF	BYTE	B#16#57 (OB57 启动请求)
OB57_PRIORITY	BYTE	组态的优先级, 默认值为2
OB57_OB_NUMBR	BYTE	OB 号 (57)
OB57_RESERVED_1	BYTE	备用
OB57_IO_FLAG	BYTE	输入模板: B#16#54 输出模板: B#16#55
OB57_MDL_ADDR	WORD	中断触发组件 (模板) 的逻辑地址
OB57_LEN	BYTE	中断提供的数据块长度
OB57_TYPE	BYTE	“属于制造厂商中断” 的中断ID 号
OB57_SLOT	BYTE	中断触发组件 (模板) 的槽号
OB57_SPEC	BYTE	详细说明 • 位0 至1: 中断详细说明 • 位 2: ADD_Ack • 位3 至7: Seq. No.
OB57_DATE_TIME	DATE_AND_TIME	OB 被调用的日期时间

注意: 您可以在OB57 中通过调用SFB54 “RALRM”, 从工业现场总线DP 信息报文中可以得到更多信息。

## 2. 9 多处理器中断组织块 (OB60)

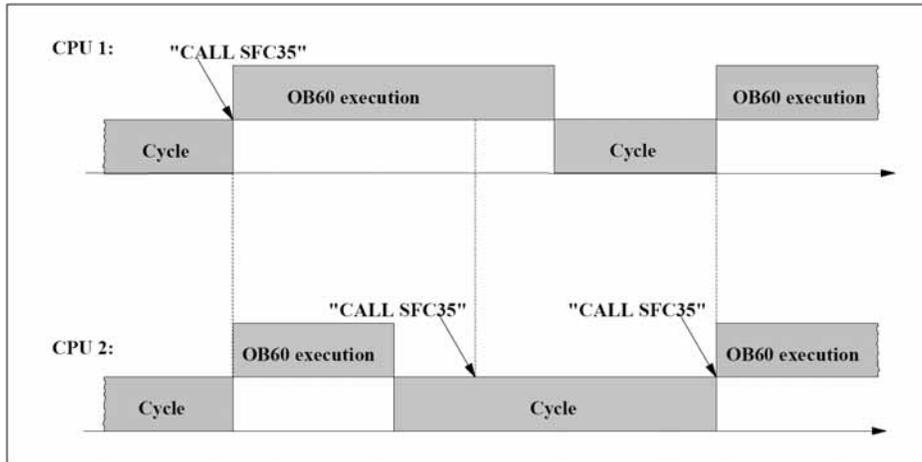
### 2. 9. 1 描述

使用多处理器中断, 您可以确定在多处理器运行期间CPU 的反应是否与一个事件同步。同硬件中断由信号模板触发相反, 多处理器中断仅由CPU 输出。

### 2. 9. 2 理解多处理器中断OB 的运行

多处理器中断通过调用SFC35 “MP\_ALM” 触发。在多处理器运行期间, 它使得总线段中的所有CPU 的OB60 同步启动, 除非您封锁OB60 (用SFC39 “DIS\_IRT”) 或延迟它 (用SFC41 “DIS\_AIRT”)。如果您没有把OB60 下装到CPU, CPU 返回中断之前的上一优先级并在那里继续程序执行。在单处理器运行时和使用分段的机架, OB60 仅在您调用SFC35 “MP\_ALM” 的CPU 中启动。

当您的程序调用SFC35 “MP\_ALM”, 您要给出任务ID。这一ID 传送到所有CPU。由此您可以对特定事件作出反应。如果您在不同的CPU 中编写不同的OB60, 这可能导致OB 的执行时间不同。在这种情况下, CPU 在不同的时间返回被中断的优先级。如果当一个CPU 仍在忙于执行前一个多处理器中断的OB60 时一个CPU 又发出多处理器中断, 那么不但发出请求的而且属于同一总线的其它所有CPU 的OB60 都不启动。两个CPU 的举例如下图所示。您可以通过所调用的SFC35 的功能数据中得到信息。



### 2. 9. 3 多处理器中断OB 的局部数据

下表描述了多处理器中断OB 的临时（TEMP）变量。变量名是OB60 的缺省名称。

变量	类型	描述
OB60_EV_CLASS	BYTE	事件代码和ID: B#16#11: 中断被激活
OB60_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>B#16#61 : 多处理器中断由本CPU 触发</li> <li>B#16#62 : 多处理器中断由其它CPU 触发</li> </ul>
OB60_PRIORITY	BYTE	所选的优先级: 缺省25
OB60_OB_NUMBR	BYTE	OB 号: 60
OB60_RESERVED_1	BYTE	备用
OB60_RESERVED_2	BYTE	备用
OB60_JOB	INT	任务ID: SFC35 “MP_ALM” 的输入变量 JOB
OB60_RESERVED_3	INT	备用
OB60_RESERVED_4	INT	备用
OB60_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

## 2. 10 周期同步中断OB (OB 61 至OB 64)

### 2. 10. 1 描述

周期同步中断给您提供与DP 周期同步启动程序的选择。OB61 作为接口OB 提供周期同步中断TSAL1。您可以在0（不选择OB）和从2 至26 之间为OB 61 设定优先级。

### 2. 10. 2 周期同步中断OB 的局部数据

下表描述了周期同步中断OB 的临时（TEMP）变量。变量名为OB 61 的缺省名称。

变量	类型	描述
OB61_EV_CLASS	BYTE	事件代码和ID: B#16#11: 中断被激活
OB61_STRT_INF	BYTE	B#16#64: OB 61 的启动请求 : B#16#67: OB 64 的启动请求
OB61_PRIORITY	BYTE	选择的优先级; 缺省: 25
OB61_OB_NUMBR	BYTE	OB 号: 61 ... 64

OB61_RESERVED_1	BYTE	备用
OB61_RESERVED_2	BYTE	备用
OB61_GC_VIOL	BOOL	GC 违反
OB61_FIRST	BOOL	启动或停止状态后首次使用
OB61_MISSED_EXEC	BYTE	从OB 61 最后一次执行以来OB61 启动失败的次数
OB61_DP_ID	BYTE	同步DP 主站系统的DP 主站系统ID

变量	类型	描述
OB61_RESERVED_3	BYTE	备用
OB61_RESERVED_4	WORD	备用
OB61_RESERVED_5	WORD	备用
OB61_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

## 2. 11 I/O 冗余故障OB (OB70)

注意：I/O 冗余故障OB (OB70) 仅可用在H CPU。

### 2. 11. 1 描述

当PROFIBUS DP 上的冗余丢失（例如，主动的DP 主站总线故障或DP 从站的接口模板出现故障时）或当主动的DP 主站的DP 从站连接的I/O 改变时，H CPU 的操作系统调用OB70。

如果启动事件出现且OB70 未编程时CPU 不变为STOP 方式。如果OB70 已下装且如果该H 系统处于冗余方式，两个CPU 都执行OB70。H 系统保持冗余方式。

### 2. 11. 2 I/O 冗余OB 的局部数据

下表含有I/O 冗余故障OB 的临时 (TEMP) 变量。变量名为OB70 的缺省名称。

变量	类型	描述
OB70_EV_CLASS	BYTE	事件级别和ID： <ul style="list-style-type: none"> <li>• B#16#72：离去事件</li> <li>• B#16#73：到来事件</li> </ul>
OB70_FLT_ID	BYTE	故障代码（可能值：B#16#A2, B#16#A3）
OB70_PRIORITY	BYTE	优先级：可通过STEP7 选择（硬件组态）
OB70_OB_NUMBR	BYTE	OB 号（70）
OB70_RESERVED_1	WORD	备用
OB70_INFO_1	WORD	根据故障代码
OB70_INFO_2	WORD	根据故障代码
OB70_INFO_3	WORD	根据故障代码
OB70_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

下表显示OB70 启动时的事件。

OB70_FLT_ID	OB70 的启动事件
B#16#A2	DP 主站或DP 主站系统故障
B#16#A3	冗余丢失/DP 从站的冗余恢复

根据故障代码这些变量有以下意义：

故障代码	位	意义
B#16#A2 OB70_INFO_1: OB70_INFO_2: OB70_INFO_3:	0 至7: 8 至15:	出问题的DP 主站的逻辑起始地址 备用 备用 出问题的DP 主站的DP 主站系统ID
B#16#A3 OB70_INFO_1: OB70_INFO_2:  OB70_INFO_3	0 至14:  15: 0 至7:  8 至15:	DP 主站的逻辑起始地址 受影响的DP 从站: 逻辑起始地址, 如果使用S7 从站 或, 诊断地址, 如果使用DP 标准 从站。 I/O 标识受影响的DP 从站DP 站 的数量 DP 主站系统ID

注意：如果您使用具有DPV1 能力的CPU，您可以借助于SFB54 “RALRM” 获得超出OB 启动信息的进一步的中断信息。当您以S7 兼容方式使用DP 主站时也是一样。

## 2. 12 CPU 冗余故障OB (OB72)

注意：CPU 冗余故障OB (OB72) 仅用于H CPU。

### 2. 12. 1 描述

当下列事件出现时H CPU 的操作系统调用OB72:

- CPU 冗余丢失
- 预定的主站切换
- 同步故障
- SYNC 模板故障
- 刷新终止
- 比较故障 (例如, RAM, PIQ)

所有处于RUN 或STARTUP 方式的CPU 随着相应的启动事件执行OB72。

### 2. 12. 2 CPU 冗余故障OB 的局部数据

下表含有CPU 冗余故障OB 的临时 (TEMP) 变量。变量名为OB72 的缺省名。

变量	类型	描述
OB72_EV_CLASS	BYTE	事件级别和ID: • B#16#78: 离去事件 • B#16#73, B#16#75, B#16#79 : 到来事件
OB72_FLT_ID	BYTE	故障代码 (可能值: B#16#01, B#16#02, B#16#03, B#16#20, B#16#21, B#16#22, B#16#23, B#16#31, B#16#33, B#16#34, B#16#35, B#16#40, B#16#41, B#16#42, B#16#43, B#16#44, B#16#50, B#16#51,

		B#16#52, B#16#53, B#16#54, B#16#55, B#16#56, B#16#C1, B#16#C2)
OB72_PRIORITY	BYTE	优先级: 可通过STEP 7 选择 (硬件组态)
OB72_OB_NUMBR	BYTE	OB 号 (72)
OB72_RESERVED_1	WORD	仅用于故障代码B#16#03: <ul style="list-style-type: none"> <li>高字节: OB72_INFO_2 和OB72_INFO_3 内容的ID - 0: OB72_INFO_2 和OB72_INFO_3 没意义- B#16#C4 : 伴随着备用主站切换 (如果 OB72_INFO_3=W#16#0001) 或不伴随备用主站切换 (如果OB72_INFO_3=W#16#0002) 故障排除之后转换到冗余方式。</li> <li>OB72_INFO_2 预留。- B#16#CD: OB72_INFO_2 和OB72_INFO_3 含有优先级 &gt;15 的实际封锁时间</li> <li>低字节: 预留</li> </ul>

变量	类型	描述
OB72_INFO_1	WORD	仅用于故障代码B#16#C2: <ul style="list-style-type: none"> <li>高字节: 超出监视时间的ID: - 1: 扫描周期时间增加- 2: I/O 无效时间- 3: 通讯时间延迟</li> <li>低字节: 当前的刷新尝试</li> </ul>
OB72_INFO_2	WORD	仅用于故障代码B#16#03 且 OB72_RESERVED_1= B#16#CD: 以ms 优先级>15 的实际封锁时间的高字节
OB72_INFO_3	WORD	仅用于故障代码B#16#03: <ul style="list-style-type: none"> <li>OB72_RESERVED_1=B#16#C4: - W#16#0001: 备用主站切换, 排除了故障之后转换到冗余方式- W#16#0002: 没有备用主站的切换, 排除了故障之后转换到冗余方式</li> <li>OB72_RESERVED_1=B#16#C4 : 优先级 &gt;15 的以ms 实际封锁时间的高字节</li> </ul>
OB82_DATE_TIME	DATE_AND_TIME	OB被调用的日期和时间

下表显示导致OB72 启动的事件。

OB72_FLT_ID	OB72 的启动事件
B#16#01	由于CPU 故障冗余丢失 (二中之二)
B#16#02	由于用户触发STOP 而冗余丢失 (二中之二)
B#16#03	H 系统 (二中之二) 变为冗余方式
B#16#20	RAM 比较故障
B#16#21	比较过程映像输出值故障
B#16#22	比较标志、定时器或计数器故障

B#16#23	识别出不同的操作系统数据
B#16#31	由于主站故障备用主站切换
B#16#33	在运行时由于系统改变备用主站切换
B#16#34	由于同步模切连接问题备用主站切换
B#16#35	由90“H_CTRL”触发备用主站切换
B#16#40	由于超出等待时间用户程序中的同步故障
B#16#41	由于在不同的同步点等待用户程序中的同步故障
B#16#42	由于在不同的同步点等待操作系统中的同步故障
B#16#43	由于超出等待时间操作系统中的同步故障
B#16#44	由于错误的的数据操作系统中的同步故障
B#16#50	无SYNC 模块
B#16#51	未通电SYNC 模块改变
B#16#52	SYNC 模块移出/插入
B#16#53	未复位SYNC 模块改变
B#16#54	SYNC 模块：机架号选择两次
B#16#55	SYNC 模块故障/排除
B#16#56	SYNC 模块上非法机架号设置
B#16#C1	刷新中止
B#16#C2	在几次尝试期间 (1 ≤ n ≤ 由于监视时间超出中止后刷新尝试的最大可能数) 因为监视时间超出刷新尝试中止

## 2. 13 通讯冗余故障OB (OB73)

注意：通讯冗余故障OB (OB73) 仅对CPU417-4H 硬件版本V2.0 有效。

### 2. 13. 1 描述

当首次冗余丢失出现在容错S7 连接时 (容错S7 连接仅用于S7 通讯。更多信息, 参阅“S7-400H 可编程控制器, 容错系统”), H CPU 操作系统调用OB73。如果冗余丢失出现在以后的容错S7 连接时, 不再有OB73 启动。直到所有容错S7 连接的冗余恢复之前, OB73 都不会再次启动。如果启动事件发生且OB73 未编程, CPU 不变为STOP 方式。

### 2. 13. 2 CPU 冗余故障OB 的局部数据

下表含有通讯冗余故障OB 的临时 (TEMP) 变量。OB73 的缺省名用做变量名。

变量	类型	描述
OB73_EV_CLASS	BYTE	事件代码和ID: B#16#73, B#16#72
OB73_FLT_ID	BYTE	故障代码 (可能值: B#16#E0)
OB73_PRIORITY	BYTE	选择的优先级: 缺省25
OB73_OB_NUMBR	BYTE	OB 号 (73)
OB73_RESERVED_1	WORD	保留
OB73_INFO_1	WORD	(与用户无关)
OB73_INFO_2	WORD	(与用户无关)

OB73_INFO_3	WORD	(与用户无关)
OB73_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

下表显示导致OB73 启动的事件。

OB73_FLT_ID	OB73 的启动事件 B#16#E0	通讯中冗余丢失/故障排除
-------------	-----------------------	--------------

## 2. 14 时间故障组织块 (OB80)

### 2. 14. 1 描述

OB 执行时出现故障S7-300 CPU 的操作系统调用OB80。这样的故障包括：循环时间超出、执行OB 时应答故障、向前移动时间以致于跃过了OB 的启动时间、CiR 后恢复RUN 方式。如果，例如，当循环中断OB 仍在执行前一次调用时，该OB 块的启动事件发生，操作系统调用OB80。如果OB80 未编程，CPU 变为STOP 方式。您可以使用SFC39 至42 封锁或延时和再使能时间故障OB。

注意：如果在同一个扫描周期中由于扫描时间超出OB80 被调用两次，CPU 就变为STOP 方式。您可以通过在程序中适当的位置调用SFC43 “RE\_TRIGR ” 来避免这种情况。

### 2. 14. 2 时间故障OB 的局部数据

下表描述了时间故障OB 的临时 (TEMP) 变量。变量名是OB80 的缺省名。

变量	类型	描述
OB80_EV_CLASS	BYTE	事件级别和标识: B#16#35
OB80_FLT_ID	BYTE	故障代码: (允许值: B#16#01, B#16#02, B#16#05, B#16#06, B#16#07, B#16#08, B#16#09, B#16#0A)
OB80_PRIORITY	BYTE	优先级: 在RUN 方式时OB80 以优先级26 运行, OB 请求缓冲区溢出时以优先级28 运行
OB80_OB_NUMBR	BYTE	OB 号 (80)
OB80_RESERVED_1	BYTE	保留
OB80_RESERVED_2	BYTE	保留
OB80_ERROR_INFO	WORD	故障信息: 根据故障代码
OB80_ERR_EV_CLAS	BYTE	引起故障的启动事件的事件级别
OB80_ERR_EV_NUM	BYTE	引起故障的启动事件的事件号
OB80_OB_PRIORITY	BYTE	故障信息: 根据故障代码
OB80_OB_NUM	BYTE	故障信息: 根据故障代码
OB80_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

根据故障代码这些变量有以下意义:

故障代码	位	意义
B#16#01		循环时间超出。
OB80_ERROR_INFO:		最后一个扫描周期的运行时间 (ms)。

OB80_ERR_EV_CLASS:		触发中断的事件的级别。
OB80_ERR_EV_NUM:		触发中断的事件的号码。
OB80_OB_PRIORITY:		当故障出现时正在执行的OB 的优先级。
OB80_OB_NUM		当故障出现时正在执行的OB 的号码。
B#16#02		调用的OB 仍在执行。
OB80_ERROR_INFO:		由OB80_ERR_EV_CLASS 和
		OB80_ERR_EV_NUM
		确定的调用块的有关临时变量。
OB80_ERR_EV_CLASS:		触发中断的事件的级别。
OB80_ERR_EV_NUM:		触发中断的事件的号码。
OB80_OB_PRIORITY		引起故障的OB 的优先级（例如：“7”指OB30/ 优先级7 它应该启动，但不能启动）。
OB80_OB_NUM:		引起故障的OB 的号码（例如：“30”指OB30 它应该启动，但不能启动。）
B#16#05 和		由于向前移动时间日期跳过时间中断。
B#16#06		HOLD 之后返回RUN 日期跳过时间中断。
OB80_ERROR_INFO:	位0 置位:	日期时间中断0 的启动时间在过去。
	位7 置位:	日期时间中断7 的启动时间在过去。
	位8 至15:	未用
OB80_ERR_EV_CLASS:		未用
OB80_ERR_EV_NUM:		未用
OB80_OB_PRIORITY:		未用
OB80_OB_NUM:		未用
B#16#07		当前优先级的OB 请求缓冲区溢出。
参数的意义参见故障代码		一个优先级的每个OB 启动请求将输入到相应的
B#16#02。		OB 请求缓冲区，OB 完成之后该输入被删除。如果一个优先级的OB 启动请求多于相应的OB 请求
		缓冲区中最大允许输入数，OB80 将被调用且故障代码为B#16#07）。
B#16#08		同步循环中断时间故障
参数的意义参见故障代码		
B#16#02。		
B#16#09		由于高中断装载中断丢失

参数的信息参见故障代码		
B#16#02		
B#16#0A		
OB80_ERROR_INFO:		CiR 之后恢复RUN
		CiR 同步时间, 以ms 为单位

## 2. 15 电源故障组织块 (OB81)

### 2. 15. 1 描述

与电源（仅对S7-400）或后备电池（当到来和离去事件）有关的故障事件发生时，S7-300CPU 的操作系统调用OB81。在S7-400 中，如果电池测试功能已通过BATT. INDIC 开关激活，仅在电池故障事件发生时调用OB81。

如果OB81 未编程，CPU 并不转换为STOP 方式。您可以用SFC39 至42 来禁用、延时或再使能电源故障OB。

### 2. 15. 2 电源故障OB 的局部数据

下表描述了电源故障OB 的临时 (TEMP) 变量。变量名是OB81 的缺省名。

变量	类型	描述
OB81_EV_CLASS	BYTE	事件级别和标识: B#16#38: 离去事件 B#16#39: 到来事件
OB81_FLT_ID	BYTE	故障代码 (可能值) B#16#21, B#16#22, B#16#23, B#16#25, B#16#26, B#16#27, B#16#31, B#16#32, B#16#33)
OB81_PRIORITY	BYTE	• 优先级; 可通过STEP 7 选择 (硬件组态) • 例如, RUN 方式的可能值: 2-26
OB81_OB_NUMBR	BYTE	OB 号 (81)
OB81_RESERVED_1	BYTE	保留
OB81_RESERVED_2	BYTE	保留
OB81_MDL_ADDR	INT	• 位0 至2: 机架号 • 位3: 0=备用CPU, 1=主站CPU • 位4 至7: 1111
OB81_RESERVED_3	BYTE	仅与故障代码B#16#31, B#16#32 and B#16#33 有关
OB81_RESERVED_4	BYTE	
OB81_RESERVED_5	BYTE	
OB81_RESERVED_6	BYTE	
OB81_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

变量OB81\_RESERVED\_I,  $3 \leq i \leq 6$  显示扩展机架上后备电池 (故障代码B#16#31)、后备电压 (故障代码 B#16#32) 或24V 电源 (故障代码B#16#33) 故障。下表显示变量OB81\_RESERVED\_I,  $3 \leq i \leq 6$  哪位对应哪个扩展机架。变量OB81\_RESERVED\_I 中的位有下述意义 (对相关的扩展机架):

	OB81_RESERVED_6	OB81_RESERVED_5	OB81_RESERVED_4	OB81_RESERVED_3
位0	保留	第8 扩展机架	第16 扩展机架	备用
位1	第1 扩展机架	第9 扩展机架	第17 扩展机架	备用
位2	第2 扩展机架	第10 扩展机架	第18 扩展机架	备用
位3	第3 扩展机架	第11 扩展机架	第19 扩展机架	备用
位4	第4 扩展机架	第12 扩展机架	第20 扩展机架	备用
位5	第5 扩展机架	第13 扩展机架	第21 扩展机架	备用
位6	第6 扩展机架	第14 扩展机架	备用	备用
位7	第7 扩展机架	第15 扩展机架	备用	备用

当事件发生时，显示扩展机架（相应位置位）电池或后备电压或24 V 电源故障。更早发生的扩展机架电池或后备电压或24 V 电源故障不再显示。事件排除且至少一个扩展机架后备电源恢复时也显示（相应位置位）。变量OB81\_FLT\_ID 有下列意义：

OB81_FLT_ID	意义
B#16#21:	中央机架的至少一个后备电池耗尽/问题排除（BATTf） 如果仅两个电源中的一个故障（如果是冗余后备电池） 事件也发生。如果 第二个电池也发生故障，事件不再出现。
B#16#22:	中央机架的后备电压故障/问题排除（BAF）
B#16#23:	中央机架的24 V 电源故障/问题排除。
B#16#25:	至少一个冗余中央机架中至少一个后备电池耗尽/问题 排除（BATTf）
B#16#26:	至少一个冗余中央机架中后备电压故障（BAF）
B#16#27:	至少一个冗余中央机架24 V 供给故障
B#16#31:	至少一个扩展机架的至少一个后备电池耗尽/问题排除 （BATTf）
B#16#32:	至少一个扩展机架的后备电压故障（BAF）
B#16#33:	至少一个扩展机架的24 V 电源故障/问题排除。

## 2. 16 诊断中断组织块（OB82）

### 2. 16. 1 描述

如果模块具有诊断能力又使能了诊断中断，当它检测到故障时，它输出一个诊断中断请求给CPU（到来和离去事件）。于是操作系统调用OB82。OB82 的

局部变量中含有故障模板四个BYTE 的诊断数据（见下表）。如果OB82 未编程，CPU 变为STOP 方式。您可以用SFC39 至42 来禁止或延时并再使能诊断中断OB。

## 2. 16. 2 诊断中断OB 的局部数据

下表描述了诊断中断OB 的临时（TEMP）变量。变量名是OB82 的缺省名。

变量	类型	描述
OB82_EV_CLASS	BYTE	事件级别和标识： • B#16#38：离去事件 • B#16#39：到来事件
OB82_FLT_ID	BYTE	故障代码（B#16#42）
OB82_PRIORITY	BYTE	• 优先级；可通过STEP 7 选择（硬件组态）
OB82_OB_NUMBR	BYTE	OB 号（82）
OB82_RESERVED_1	BYTE	备用
OB82_IO_FLAG	BYTE	• 输入模板：B#16#54 • 输出模板：B#16#55
OB82_MDL_ADDR	WORD	故障发生处模板的逻辑起始地址
OB82_MDL_DEFECT	BOOL	模板故障
OB82_INT_FAULT	BOOL	内部故障
OB82_EXT_FAULT	BOOL	外部故障
OB82_PNT_INFO	BOOL	通道故障
OB82_EXT_VOLTAGE	BOOL	外部电压故障
OB82_FLD_CONNCTR	BOOL	前连接器未插入
OB82_NO_CONFIG	BOOL	模板未组态
OB82_CONFIG_ERR	BOOL	模板参数不正确
OB82_MDL_TYPE	BYTE	• 位0 至3：模板级别 • 位4：通道信息存在 • 位5：用户信息存在 • 位6：来自替代的诊断中断 • 位7：备用
OB82_SUB_MDL_ERR	BOOL	子模板丢失或有故障
OB82_COMM_FAULT	BOOL	通讯问题
OB82_MDL_STOP	BOOL	操作方式（0：RUN，1：STOP）

变量	类型	描述
OB82_WTCH_DOG_FLT	BOOL	看门狗定时器响应
OB82_INT_PS_FLT	BOOL	内部电源故障
OB82_PRIM_BATT_FLT	BOOL	电池故障
OB82_BCKUP_BATT_FLT	BOOL	全部后备电池故障
OB82_RESERVED_2	BOOL	备用
OB82_RACK_FLT	BOOL	扩展机架故障

OB82_PROC_FLT	BOOL	处理器故障
OB82_EEPROM_FLT	BOOL	EPROM 故障
OB82_RAM_FLT	BOOL	RAM 故障
OB82_ADU_FLT	BOOL	ADC/DAC 故障
OB82_FUSE_FLT	BOOL	熔断器熔断
OB82_HW_INTR_FLT	BOOL	硬件中断丢失
OB82_RESERVED_3	BOOL	备用
OB82_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

注意：如果您有DPV1 能力的CPU 借助SFB54 您可以得到超出OB 启动信息的中断的进一步信息。当您以S7 兼容方式运行DP 主站时也是一样。

## 2. 17 模板插/拔中断组织块 (OB83)

### 2. 17. 1 描述

在下列情况下CPU 操作系统调用OB83：

- 组态的模板插入/拔出之后
- 在STEP 7 下修改了模板参数并在RUN 状态下装所作修改到CPU 之后您可以借助SFC39 至42 禁止/延时/使能插入/拔出中断OB。

模板插入和拔出在RUN、STOP 和STARTUP 方式时每次组态的模板插入或拔出，就产生了插入/拔出中断（电源模板、CPU、适配模块和IM 不能在这种方式下移出）。该中断引起有关CPU 的诊断缓冲区和系统状态表的记录。如果CPU 在RUN 方式也启动插入/拔出OB。如果该OB 未编程，CPU 变为STOP 方式。

系统以秒间隔查询-400 模板以检测插入或拔出。为使CPU 能检测出S7-400 模板的拔出和插入，拔出和插入之间最小时间间隔必须超出两秒。对其它模板最小时间更高一些。

如果您在RUN 方式下拔出组态的模板，OB83 启动。由于仅以一秒的间隔监视模板的存在，如果模板被直接访问或当过程映像被刷新时可能首先检测出访问故障。如果您在RUN 方式下在组态的插槽中插入一块模板，操作系统检查插入的模板类型是否与组态记录一致。如果模板类型匹配，于是OB83 被启动并且参数被赋值。

### 2. 17. 2 重新组态模板

在运行时，修改系统组态 (CiR) 可以重新选择已存在模板的参数。参数的重新选择通过传送所需的参数数据到模板来进行。过程如下：

1. 在您用STEP 7 为一块模板选择了新的参数并且在RUN 方式下下装该参数组态到CPU 后OB83 启动（启动事件W#16#3367）。相应的OB 启动信息为逻辑起始地址 (OB83\_MDL\_ADDR) 和模块类型 (OB83\_MDL\_TYPE)。模板I/O 数据现在可能不对，这意味着不可能有SFC 忙于发送数据记录到该模板。
2. OB 83 执行之后模板参数被重新赋值。
3. 参数赋值之后OB 83 将再次启动（启动事件W#16#3267，成功提供参数赋值，或W#16#3968 如果故障）。模板I/O 数据响应与插入中断后的响应相同，也就是，现在它们可能不正确。现在您可以再次调用SFC 发送数据记录到该模板。

### 2. 17. 3 OB 83 的局部数据

下表描述插入/拔出模板中断OB 的临时（TEMP）变量。变量名是OB83 的缺省名称。

变量	类型	描述
OB83_EV_CLASS	BYTE	事件级别和标识： • B#16#32：模板参数赋值结束 • B#16#33：模板参数赋值启动 • B#16#38：模板插入 • B#16#39：模板拔出或无反应，或参数赋值结束
OB83_FLT_ID	BYTE	故障代码：（可能值B#16#61，B#16#63，B#16#64，B#16#65，B#16#67，B#16#68，B#16#84）
OB83_PRIORITY	BYTE	优先级，可通过STEP 7 选择（硬件组态）
OB83_OB_NUMBR	BYTE	OB 号（83）
OB83_RESERVED_1	BYTE	块模板或接口模板标识
OB83_MDL_TD	BYTE	范围： • B#16#54：外设输入（PI） • B#16#55：外设输出（PQ）
OB83_MDL_ADDR	WORD	有关模板的逻辑起始地址
OB83_RACK_NUM	WORD	• 如果OB83_RESERVED_1 =B#16#A0：接口模板号 • 如果OB83_RESERVED_1 = B#16#C4：机架号或DP 站号（低字节）或DP 主站系统ID（高字节）

变量	类型	描述
OB83_MDL_TYPE	WORD	有关模板的模板类型： • W#16#X5XX：模拟量模板 • W#16#X8XX：功能模板 • W#16#XCXX：CP • W#16#XFXX：数字量模板 X：数值对用户无效
OB83_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

由故障代码OB83\_FLT\_ID 决定的变量OB83\_MDL\_TYPE 有下列意义：

故障代码	意义
B#16#61:  OB83_MDL_TYPE:	模板插入。模板类型正确 （对事件级别B#16#38） 模板拔出或无响应 （对事件级别B#16#39） 实际模板类型
B#16#63 OB83_MDL_TYPE:	模板插入但模板类型不正确 实际模板类型
B#16#64:	模板插入但有问题 （读不出模板ID）

OB83_MDL_TYPE:	组态模板类型
B#16#65: OB83_MDL_TYPE:	模板插入但模板参数赋值故障 实际模板类型
W#16#3866	模板再次响应, 装载电压故障改正
W#16#3966	模板不再次响应, 装载电压故障改正
W#16#3367	模板重新组态启动
W#16#3267	模板重新组态结束
W#16#3968	模板重新组态结束有故障
W#16#3884	接口模板插入
W#16#3984	接口模板拔出

注意: 如果您使用具有DPV1能力的CPU, 借助SFB54“RALRM”, 您可以得到OB启动信息之外的中断的进一步信息。当您以S7兼容方式运行DP主站时也是一样。

## 2. 18 CPU 硬件故障组织块 (OB84)

### 2. 18. 1 描述

当前CPU的OS不再调用OB84。

### 2. 18. 2 硬件故障OB的局部数据

下表含有CPU硬件故障的临时 (TEMP) 变量。变量名是OB84的缺省名。

变量	类型	描述
OB84_EV_CLASS	BYTE	事件级别和标识: • B#16#38 : 离去事件 • B#16#39 : 到来事件
OB84_FLT_ID	BYTE	故障代码 (B#16#81)
OB84_PRIORITY	BYTE	优先级, 可通过STEP 7 选择 (硬件组态)
OB84_OB_NUMBR	BYTE	OB号 (84)
OB84_RESERVED_1	BYTE	备用
OB84_RESERVED_2	BYTE	备用
OB84_RESERVED_3	WORD	备用
OB84_RESERVED_4	DWORD	备用
OB84_DATE_TIME	DATE_AND_TIME	OB被调用时的日期和时间

## 2. 19 优先级故障组织块 (OB85)

### 2. 19. 1 描述

当下列事件之一发生时, CPU的操作系统调用OB85:

- 未下装的OB的启动事件 (OB81除外)。
- 当操作系统访问模板时故障。
- 在由系统刷新过程映像期间I/O访问故障 (如果OB85调用没有在组态中禁止)。

注意: 如果OB85未编程, 当检测到这些事件之一时CPU变为STOP方式。您可以使用SFC39至42封锁或延时并使能优先级故障OB。

## 2. 19. 2 优先级故障OB 的局部数据

下表描述优先级故障OB 的临时 (TEMP) 变量。变量名是OB85 的缺省名。如果您要根据可能的故障代码编写OB85, 我们提示您按下述组织局部数据: 根据故障代码, 修改过的和缺省的变量比较有下述意义:

变量	类型	描述
OB88_EV_CLASS	BYTE	事件级别和标识: B#16#35 B#16#38 (仅有故障代码B#16#B3 和B#16#B4) B#16#39 (仅有故障代码B#16#B1 和 B#16#B2, B#16#B3 和B#16#B4)
OB85_FLT_ID	BYTE	故障代码 (可能值: B#16#A1 , B#16#A2 , B#16#A3 , B#16#B1, B#16#B2, B#16#B3, B#16#B4)
OB85_PRIORITY	BYTE	优先级, 可通过STEP 7 选择 (硬件 组态)
OB85_OB_NUMBR	BYTE	OB 号 (85)
OB85_RESERVED_1	BYTE	备用
OB85_RESERVED_2	BYTE	备用
OB85_RESERVED_3	INT	备用
OB85_ERR_EV_CLASS	BYTE	引起故障的事件级别
OB85_ERR_EV_NUM	BYTE	引起故障的事件号码
OB85_OB_PRIOR	BYTE	当故障发生时被激活的OB 的优先级
OB85_OB_NUM	BYTE	当故障发生时被激活的OB 的号码
OB85_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间
OB85_EV_CLASS		BYTE
OB85_FLT_ID		BYTE
OB85_PRIORITY		BYTE
OB85_OB_NUMBR		BYTE
OB85_DKZ23		BYTE
OB85_RESERVED_2		BYTE
OB85_Z1		WORD
OB85_Z23		DWORD
OB85_DATE_TIME		DATE_AND_TIME

故障代码	字节/字	意义
B#16#A1 和 B#16#A2 OB85_Z1: OB85_Z23:	高字: 低字: 高 字节/ 低字节	用STEP 7 生成的组态结果, 您的程序 或操作系统使用的OB 未下装到CPU 的 启动事件。用STEP 7 生成的组态结果, 您的程序或操作系统产生的OB 未下装 到CPU 的启动事件。调用的OB 的相关 局部变量由OB85_Z23 决定。引起OB 调 用的事件的级别和号码故障时的程序 级和激活的OB
B#16#A3 OB85_Z1:  OB85_Z23	高字节:  低字节:  高字:  低字:	当操作系统访问模块时故障 操作系统的故障ID 1: 集成功能 2: IEC 定时器 0: 无故障分辨能力 1: 块未装载 2: 区域长度故障 3: 写保护故障 块号 引起故障的MC7 命令的相对地址。块 类型必须从 OB85_DKZ23 得到 (B#16#88: OB, B#16#8C: FC, B#16#8E: FB, B#16#8A: DB)
B#16#B1 和 B#16#B2:		当刷新输入的过程映像时I/O 访问故 障 当传送输出的过程映像到输出模板时 I/O 访问故障
OB85_DKZ23:		I/O 访问故障发生时过程映像传送类 型的ID • B#16#10 : 字节访问 • B#16#20 : 字节访问 • B#16#30 : 双字访问 • B#16#57 : 传送组态的一致性区域
OB85_Z1:		由CPU 内部使用占用: 模板的逻辑基 地址 如果OB85_RESERVED_2 有数值 B#16#76, OB85_Z1 接收有关SFC (SFC 14, 15, 26 和27) 的返回值。
OB85_Z23:	字节0:	过程映像区号。
	字节1:	• 无效, 如果OB85_DKZ23=B#16#10, 20 或30 • 一致性区域长度, 单位字节, 如果

		OB85_DKZ23 = B#16#57
	字节2 和3:	<ul style="list-style-type: none"> <li>PII 引起的I/O 地址, 如果 OB85_DKZ23= B#16#10、20 或30</li> <li>一致性区域的逻辑起始地址, 如果 OB85_DKZ23 = B#16#57</li> </ul>

故障代码	字节/字	意义
如果您已经组态了当系统过程映像表刷新时发生I/O 访问故障重复调用OB85，您将获得故障代码B#16#B1 和B#16#B2。		
B#16#B3: B#16#B4:		当刷新输入的过程映像时I/O 访问故障，到来/离去事件当传送输出过程映像至输出模板时I/O 访问故障，到来/离去事件
OB85_DKZ23:		I/O 访问故障发生时过程映像传送类型的ID • B#16#10：字节访问 • B#16#20：字访问 • B#16#30：双字访问 • B#16#57：传送组态的一致性区域
OB85_Z1:		由CPU 内部占用：模板的逻辑起始地址如果OB85_RESERVED_2 有数值 B#16#76，OB85_Z1 为与接收SFC（SFC 14, 15, 26 和27）有关的返回值。
OB85_Z23:	字节0:	过程映像区号。
	字节1:	<ul style="list-style-type: none"> <li>无效, 如果OB85_DKZ23=B#16#10, 20 或30</li> <li>一致性区域的长度, 单位字节, 如果OB85_DKZ23= B#16#57</li> </ul>
	字节2 和3:	<ul style="list-style-type: none"> <li>PII 引起的I/O 地址, 如果 OB85_DKZ23= B#16#10, 20 或30</li> <li>连续区域的逻辑起始地址, 如果 OB85_DKZ23= B#16#57</li> </ul>
如果您已经组态了在系统刷新过程映像表期间如果I/O 访问故障到来和离去事件发生时调用OB85，您将获得故障代码B#16#B3 和B#16#B4。在冷或暖启动之后，在下一次过程映像表刷新时，所有不存在的输入和输出的访问都将作为I/O 访问故障报告。		

## 2. 20 机架故障组织块（OB86）

### 2. 20. 1 描述

当扩展机架（不是CPU 318）、DP 主站系统、或分布式I/O 中从站故障时（到来和离去事件时），CPU 的操作系统调用OB86。如果OB86 未编程，当检测到

这类故障时，CPU 进入STOP 方式。您可以使用SFC 39 至42 禁止、延时或使能OB86。

## 2. 20. 2 机架故障OB 的局部数据

下表描述机架故障OB 的临时（TEMP）变量。变量名为OB86 的缺省名。

变量	类型	描述
OB86_EV_CLASS	BYTE	事件级别和标识：• B#16#38：离去事件 • B#16#39：到来事件
OB86_FLT_ID	BYTE	故障代码：（可能值B#16#C1，B#16#C2，B#16#C3，B#16#C4，B#16#C5，B#16#C6，B#16#C7，B#16#C8）
OB86_PRIORITY	BYTE	优先级，可通过STEP 7 选择（硬件组态）
OB86_OB_NUMBR	BYTE	OB 号（86）
OB86_RESERVED_1	BYTE	备用
OB86_RESERVED_2	BYTE	备用
OB 86_MDL_ADDR	WORD	根据故障代码
OB86_RACKS_FLTD	BOOL ARRAY [0 .. 31]	根据故障代码
OB86_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

如果您要根据可能的故障代码编写OB86，我们建议您按下述组织局部数据：根据故障代码，变量的内容有下列意义：

变量	类型
OB86_EV_CLASS	BYTE
OB86_FLT_ID	BYTE
OB86_PRIORITY	BYTE
OB86_OB_NUMBR	BYTE
OB86_RESERVED_1	BYTE
OB86_RESERVED_2	BYTE
OB86_MDL_ADDR	WORD
OB86_Z23	DWORD
OB86_DATE_TIME	DATE_AND_TIME

故障代码	意义
------	----

B#16#39C1: OB86_MDL_ADDR: OB86_Z23:	扩展机架故障IM 的逻辑起始地址含有每个可能的扩展机架的一个位: 位0: 总为0 位1: 第1 扩展机架: 位21: 第21 扩展机架位22 至29: 总为0 位30: 在SIMATIC S5 区中至少一个扩展机架故障位31: 总为0
B#16#38C1:	扩展机架重新运行

意义: 当事件发生时, 报告导致OB86 调用的扩展机架故障(对应它们的位被置位)。更早发生故障的扩展机架不再显示。当故障修复时, 再次激活的扩展机架在故障代码中报告(对应它们的位被置位。)

故障代码	意义
B#16#C2: OB86_MDL_ADDR: OB86_Z23:	扩展机架恢复并且设定和实际组态不一致。IM 的逻辑起始地址含有对应每一可能扩展机架的一位, 见故障代码B#16#C1。• 一个位置位时的意义(有关的扩展机架): - 不正确类型ID 的模板存在- 组态的模板丢失- 至少一块模板有故障
B#16#C3: OB86_MDL_ADDR: OB86_Z23:	分布式I/O: 主站系统故障。(仅到来事件引起OB86 启动, 故障代码为B#16#C3。离去事件启动OB86, 故障代码B#16#C4, 事件级别B#16#38。每个DP 从站恢复启动OB86)。DP 主站的逻辑起始地址。DP 主站系统ID 位0 至7: 备用位8 至15: DP 主站系统ID 位16 至31: 备用
B#16#C4: B#16#C5:  OB86_MDL_ADDR: OB86_Z23:	DP 站故障DP 站有问题DP 主站的逻辑起始地址。  有关的DP 从站地址: 位0 至7: DP 站号位8 至15: DP 主站系统ID 位16 至30: S7 从站的逻辑起始地址或标准DP 从站的诊断地址位31: I/O 标识

故障代码	意义
B#16#C6: OB86_MDL_ADDR: OB86_Z23:	扩展机架再次运行但模板参数赋值出错 IM 的逻辑起始地址含有对应每一可能扩展机架的一位: 位0: 总为0  位1: 第1 个扩展机架

	: 位21: 第21 个扩展机架 位22 至30: 备用 位31: 总为0 当一位被置位时表示（有关扩展机架）：存在没有正确标识的模板 模板丢失或模板的参数不正确
B#16#C7: OB86_MDL_ADDR: OB86_Z23:	DP 站恢复，但模板参数赋值出错DP 主站的逻辑起始地址。相应的DP 从站的地址：位0 至7: DP 站号  位8 至15: DP 主站系统ID 位16 至30: DP 从站的逻辑起始地址 位31: I/O 标识
B#16#C8: OB86_MDL_ADDR: OB86_Z23:	DP 站恢复，设定组态与实际组态不一致DP 主站的逻辑起始地址相应的DP 从站的地址：位0 至7: DP 站号  位8 至15: DP 主站系统ID 位16 至30: DP 从站的逻辑起始地址 位31: I/O 标识

注意：

如果您使用具有DPV1 能力的CPU，借助SFB54 “RALRM”，您可以得到超出OB 启动信息的中断的进一步信息。当您以S7 兼容方式运行DP 主站时也是一样。

## 2. 21 通讯故障组织块（OB87）

### 2. 21. 1 描述

当导致通讯故障的事件发生时CPU 的操作系统调用OB87。如果OB87 未编程，CPU 不进入STOP 方式。您可以用SFC39 至42 封锁或延时并再使能通讯故障OB。

### 2. 21. 2 OB87 的局部数据

下表描述通讯故障OB 的临时（TEMP）变量。变量名为OB87 的缺省名。

变量	类型	描述
OB87_EV_CLASS	BYTE	事件级别和标识B#16#35:
OB87_FLT_ID	BYTE	故障代码：（可能值：B#16#D2, B#16#D3, B#16#D4, B#16#D5, B#16#E1, B#16#E2, B#16#E3, B#16#E4, B#16#E5, B#16#E6）
OB87_PRIORITY	BYTE	优先级，可通过STEP 7 选择（硬件组态）
OB87_OB_NUMBR	BYTE	OB 号（87）
OB87_RESERVED_1	BYTE	备用
OB87_RESERVED_2	BYTE	备用

OB87_RESERVED_3	WORD	根据故障代码
OB87_RESERVED_4	DWORD	根据故障代码
OB87_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

根据故障代码变量有下述意义:

故障代码	字节/字	意义
B#16#D2: B#16#D3: B#16#D4: B#16#D5: OB87_RESERVED_3: OB87_RESERVED_4:		当前的诊断记录传送不可能 同步信息不能传送 (主站) 由于时钟同步非法的日时间跳转。 当接收同步时间时故障 (从站)。 不含进一步信息。 不含进一步信息。
B#16#E1: B#16#E3: B#16#E4: OB87_RESERVED_3: OB87_RESERVED_4:	高字节: 低字节:	在全局数据通讯期间不正确的帧ID 在全局数据通讯期间帧长度故障 收到非法GD 包号码。 接口ID (0: K 总线, 1:MPI) GD 包号 不含进一步信息。

故障代码	字节/字	意义
B#16#E2: OB87_RESERVED_3: OB87_RESERVED_4:	高字: 低字:	GD 包状态不能记录在DB 中 DB 号 不含进一步信息。 GD 包号 (高字节) GD 包号 (低字节)
B#16#E5: OB87_RESERVED_3: OB87_RESERVED_4:	高字: 低字:	在通过通讯功能块数据交换期间对DB 访问故障 预留CPU 内部使用。 含有引起故障的MC7 命令的块的号 码。 含有引起故障的MC7 命令的相对地 址。

从OB\_87\_RESERVED\_1 可读块类型 (B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB)

故障代码	意义
B#16#E6: OB87_RESERVED_3: OB87_RESERVED_4:	GD 包状态不能记录在 DB 中 DB 号 不含进一步信息。

## 2. 22 过程中断OB (OB 88)

### 2. 22. 1 描述

在一个程序块的执行被中断后CPU 的操作系统调用OB88。这种中断可能的原因是：

- 同步故障的嵌套深度太高
- 块调用的嵌套深度 (B-堆栈) 太高
- 在局部数据定位中故障如果您未编程OB88 并且程序块执行中断, CPU 进入STOP 方式 (事件ID W#16#4570) 。如果程序块在优先级28 中执行时被中断, CPU 进入STOP 方式。您可以借助SFC39 至42 封锁、延时并使能过程中断OB。

### 2. 22. 2 过程中断OB 的局部数据

下表含有过程中断OB 的临时 (TEMP) 变量。变量名为OB88 的缺省名。

变量	类型	描述
OB88_EV_CLASS	BYTE	事件级别和ID: B#16#35
OB88_SW_FLT	BYTE	故障代码: • B#16#73 : 同步故障的嵌套深度太高• B#16#75 : 块调用的嵌套深度 (B-堆栈) 太高• B#16#76 : 局部数据定位故障
OB88_PRIORITY	BYTE	优先级: 28
OB88_OB_NUMBR	BYTE	OB 号 (88)
OB88_BLK_TYPE	BYTE	出现故障的块的类型: • B#16#88:OB • B#16#8C:FC • B#16#8E:FB • B#16##00 : 不能决定中断源
OB88_RESERVED_1	BYTE	备用
OB88_FLT_PRIORITY	BYTE	引起故障的OB 的优先级
OB88_FLT_OB_NUMBR	BYTE	引起故障的OB 的号码
OB88_BLK_NUM	WORD	引起故障的MC7 指令的块的号码
OB88_PRG_ADDR	WORD	引起故障的MC7 指令的相对地址
OB88_DATE_TIME	DATE_AND_TIME	OB 被调用的日期和时间

## 2. 23 背景组织块 (OB90)

### 2. 23. 1 描述

STEP 7 可以监视最大扫描周期时间, 并可以保证最小扫描周期时间。如果OB1 包括所有的嵌套中断和系统功能的执行时间小于您所指定的最小扫描周期时间, 操作系统反应如下:

- 它调用背景OB (如果CPU 中OB90 已存在)。
- 它延时下一次OB1 启动 (如果在CPU 中OB90 不存在)。

### 2. 23. 2 理解OB90 的运行

OB90 在所有OB 中优先级最低。它可以被任何系统功能和任务所中断 (OB1 在

最小周期时间超出时) 并仅在如果所选的最小扫描周期时间还没有达到时执行。一个例外是在OB90 中调用SFC 和SFB, 它们用OB1 优先级的执行并且不会被OB1 中断, 没有OB90 的时间监视。OB90 中的用户程序在下述情况下从第一条指令处开始处理:

- 暖、冷或热启动之后
- OB90 中正在执行的块被删除之后 (用STEP 7)
- 在RUN 方式下装OB90 到CPU 之后
- 结束了背景周期之后

注意:

当组态中最小扫描周期时间和周期监视时间之间差距不大时, 在背景OB 中 SFC 和SFB 调用可能会导致循环时间超出。

### 2. 23. 3 OB90 的局部数据

下表描述OB90 的临时 (TEMP) 变量。变量名为OB90 的缺省名。

变量	类型	描述
OB90_EV_CLASS	BYTE	事件级别和标识: B#16#11: 激活
OB90_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>• B#16#91 : 暖启动/冷启动/热启动</li> <li>• B#16#92 : 块删除</li> <li>• B#16#93 : 在RUN 方式下装OB90 至CPU</li> <li>• B#16#95 : 背景周期结束</li> </ul>
OB90_PRIORITY	BYTE	优先级: 29 (对应优先级0. 29)
OB90_OB_NUMBR	BYTE	OB 号 (90)
OB90_RESERVED_1	BYTE	备用
OB90_RESERVED_2	BYTE	备用
OB90_RESERVED_3	INT	备用
OB90_RESERVED_4	INT	备用
OB90_RESERVED_5	INT	备用
OB90_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

### 2. 24 启动的类型

启动组织块 (OB100, OB101 和 OB102)

下列启动类型是有区别的

- 热启动 (S7-300 和S7-400H 没有)
- 暖启动
- 冷启动 在下表中, 您可以看到在启动期间哪个OB 被操作系统调用。

启动类型	相应的OB
热启动	OB101
暖启动	OB100
冷启动	OB102

关于启动类型更详细的信息，参见手册“用STEP 7”编程、“用STEP 7 组态硬件和连接”和“S7-400H PLC”。

#### 启动事件

由于下列原因CPU 执行启动：

- POWER ON 之后
- 当您把方式选择开关从STOP 拨到RUN 时
- 使用通讯功能请求之后（来自编程装置的手动命令或在另一CPU 上通过调用通讯功能块19“START ”或21“RESUME”。）
- 多处理器的同步
- H 系统中连接建立之后（仅指备用的CPU）

根据启动事件，特定的CPU，和它的参数，相应的启动OB（OB100，OB101，或OB102）被调用。通过编程，您可以对您的循环程序做某些设置（例外：在H 系统，当备用CPU 连接上时，备用的CPU 启动但不调用启动OB）。

#### 2. 24. 1 启动OB 的局部数据

下表描述启动OB 的临时（TEMP）变量。

变量名为OB100 的缺省名。

变量	类型	描述
OB10_EV_CLASS	BYTE	事件级别和标识：B#16#13：激活
OB10x_STRTUP	BYTE	启动请求： <ul style="list-style-type: none"> <li>• B#16#81： 手动暖启动</li> <li>• B#16#82： 自动暖启动</li> <li>• B#16#83： 手动热启动请求</li> <li>• B#16#84： 自动热启动请求</li> <li>• B#16#85： 手动冷启动请求</li> <li>• B#16#86： 自动冷启动请求</li> <li>• B#16#87： 主站：手动冷启动请求</li> <li>• B#16#88： 主站：自动冷启动请求</li> <li>• B#16#8A： 主站：手动暖启动请求</li> <li>• B#16#8B： 主站：自动暖启动请求</li> <li>• B#16#8C： 备用：手动启动请求</li> <li>• B#16#8D： 备用：自动启动请求</li> </ul>
OB10x_PRIORITY	BYTE	优先级：27
OB10x_OB_NUMBR	BYTE	OB 号（100，101，或102）
OB10x_RESERVED_1	BYTE	备用
OB10x_RESERVED_2	BYTE	备用
OB10x_STOP	WORD	引起CPU 停机事件的号码
OB10x_STRT_INFO	DWORD	关于当前启动的进一步信息
OB10x_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

下表显示变量OB100\_STR\_INFO 和OB101\_STR\_INFO 。 1-42

位号	意义	可能的二进制值	解释
31 - 24	启动信息	0000 xxxx	机架号0 (仅H CPU)
		0100 xxxx	机架号1 (仅H CPU)
		1000 xxxx	机架号2 (仅H CPU)
		0001 xxxx	多处理器 (仅S7-400)
		0010 xxxx	这段机架超过一个CPU 在运行 (仅S7-400)
		xxxx xxx0	设定和实际组态一致 (仅S7-300)
		xxxx xxx1	设定和实际组态不一致 (仅S7-300)
		xxxx xx0x	设定和实际组态一致
		xxxx xx1x	设定和实际组态不一致
		xxxx x0xx	不是H CPU
		xxxx x1xx	H CPU
		xxxx 0xxx	在最后POWER ON 时时间标时时钟不是由电池支持的

位号	意义	可能的二进制值	解释
		xxxx 1xxx	在最后POWER ON 时时钟是由电池支持的
23 - 16	启动刚完成	0000 0001	根据参数赋值未改变CPU 上设置的多处理器暖启动 (仅S7-400)
		0000 0011	方式选择开关触发的启动 (暖)
		0000 0100	通过MPI 由命令触发的启动 (暖)
		0000 0101	根据参数赋值未改变CPU 上设置的多处理器冷启动 (仅S7-400)
		0000 0011	方式选择开关触发的冷启动
		0000 1000	通过MPI 由命令触发的冷启动
		0000 1010	根据参数赋值未改变CPU 上设置的多处理器热启动 (仅S7-400)
		0000 1011	方式选择开关触发的热启动 (仅S7-400)
		0000 1100	通过MPI 由命令触发的热启动 (仅S7-400)
		0001 0000	有电池支持POWER ON 之后的自动启动 (热)
		0001 0001	根据参数赋值有电池支持POWER ON 之后的冷启动

		0001 0011	方式选择开关触发的启动(暖), 最后POWER ON 时电池支持
		0001 0100	通过MPI 由命令触发的启动(暖), 最后POWER ON 时有电池支持
		0010 0000	有电池支持POWER ON 之后(由系统存储器复位)执行了自动启动(暖)
		0010 0001	有电池支持POWER ON 之后(由系统存储器复位)执行了冷启动
		0010 0011	方式选择开关触发的启动(暖), 最后POWER ON 时无电池支持
		0010 0100	通过MPI 由命令触发的启动(暖), 最后POWER ON 时无电池支持
		1010 0000	根据参数赋值有电池支持POWER ON 之后的热启动(仅S7-400)
15 - 12	允许自动启动吗	0000	自动启动非法, 需存储器复位
		0001	自动启动非法, 需修改参数等
		0111	自动启动允许(暖)
		1111	自动启动允许(暖/热)(仅S7-400)
11 - 8	允许手动启动吗	0000	启动非法, 需存储器复位
		0001	启动非法, 需修改参数等
		0111	启动允许(暖)

位号	意义	可能的二进制值	解释
		1111	启动允许(暖/热)(仅S7-400)
7 - 0	最后有效的干涉或POWER ON 后自动启动的设置	0000 0000	无启动
		0000 0001	根据参数赋值在CPU 上的设置无改变的多处理器暖启动(仅S7-400)
		0000 0011	方式选择开关触发的启动(暖)
		0000 0100	通过MPI 由命令触发的启动(暖)
		0000 0101	根据参数赋值在CPU 上的设置无改变的多处理器热启动(仅S7-400)
		0000 0111	方式选择开关触发的冷启动
		0000 1000	通过MPI 由命令触发的冷启动

		0000 1010	根据参数赋值在CPU 上的设置无改变的多处理器热启动（仅S7-400）
		0000 1011	方式选择开关触发的热启动（仅S7-400）
		0000 1100	通过MPI 由命令触发的热启动（仅S7-400）
		0001 0000	在电池后备POWER ON 之后自动启动（暖）
		0001 0001	根据参数赋值有电池支持POWER ON 之后的冷启动
		0001 0011	方式选择开关触发的启动（暖），最后POWER ON 时有电池支持
		0001 0100	通过MPI 由命令触发的启动（暖），最后POWER ON 时有电池支持
		0010 0000	有电池支持POWER ON 之后自动启动（暖）（由系统复位存储器）
		0010 0001	根据参数赋值有电池支持POWER ON 之后的冷启动
		0010 0011	方式选择开关触发的启动（暖），最后POWER ON 时无电池支持
		0010 0100	通过MPI 由命令触发的启动（暖），最后POWER ON 时无电池支持
		1010 0000	根据参数赋值有电池支持POWER ON 之后自动热启动（仅S7-400）

## 2. 25 编程故障组织块（OB121）

### 2. 25. 1 描述

当有关程序处理的故障事件发生时CPU 的操作系统调用OB121。例如，如果程序调用了CPU 中不存在的块，OB121 就被调用。

### 2. 25. 2 理解编程故障OB 的运行

OB121 与被中断的块在同一优先级中执行。如果OB121 未编程，CPU 从RUN 方式进入STOP 方式。S7 提供了下列SFC 在您的程序执行期间屏蔽和解除屏蔽OB121 的启动事件：

- SFC36（MSK\_FLT）：屏蔽特定的故障代码
- SFC37（DMSK\_FLT）：解除被SFC36 屏蔽的故障代码
- SFC38（READ\_ERR）：读故障寄存器

### 2. 25. 3 编程故障OB 的局部数据

下表描述编程故障OB 的临时（TEMP）变量。变量名为OB121 的缺省名。根据故障代码变量有下述意义：

变量	类型	描述
----	----	----

OB121_EV_CLASS	BYTE	事件级别和标识: B#16#25
OB121_SW_FLT	BYTE	故障代码: (可能值: B#16#21, B#16#22, B#16#23, B#16#24, B#16#25, B#16#26, B#16#27, B#16#28, B#16#29, B#16#30, B#16#31, B#16#32, B#16#33, B#16#34, B#16#35, B#16#3A, B#16#3C, B#16#3D, B#16#3E, B#16#3F)
OB121_PRIORITY	BYTE	优先级=出现故障的OB 的优先级
OB121_OB_NUMBR	BYTE	OB 号 (121)
OB121_BLK_TYPE	BYTE	出现故障的块的类型 (在S7-300 时无有效值在这里记录): B#16#88: OB, B#16#8A: DB, B#16#8C: FB, B#16#8E: FB)
OB121_RESERVED_1	BYTE	备用
OB121_FLT_REG	WORD	故障源 (根据故障代码)。例如: • 转换故障发生的寄存器 • 不正确的地址 (读/写故障) • 不正确的定时器/计数器/块号码 • 不正确的存储器区
OB121_BLK_NUM	WORD	引起故障的MC7 命令的块的号码 (S7-300 无效)
OB121_PRG_ADDR	WORD	引起故障的MC7 命令的块的号码 (S7-300 无效)
OB121_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间

故障代码	意义
B#16#21: OB121_FLT_REG:	BCD 转换故障 有关的寄存器的ID (W#16#0000: 累加器1)
B#16#22: B#16#23: B#16#28: B#16#29:  OB121_FLT_REG:	读时区域长度故障 写时区域长度故障 用指针读访问字节、字、或双字时位地址不为0。 用指针写访问字节、字、或双字时位地址不为0。 不正确的字节地址。可以从OB121_RESERVED_1 中读出数据区和访问类型。 位7 至4 访问类型。0: 位访问 1: 字节访问 2: 字访问 3: 双字访问

OB121_RESERVED_1:	位3 至0 存储器区: 0: I/O 区1: 过程映像输入表  2: 过程映像输出表 3: 位存储器 4: 全局DB 5: 背景DB 6: 自己的局部数据 7: 调用者的局部数据
B#16#24: B#16#25: OB121_FLT_REG:	当读时区间故障 当写时区间故障 在低字节含有非法区的ID (B#16#86 自己的局部数据区)
B#16#26: B#16#27: OB121_FLT_REG:	定时器号码故障 计数器号码故障 非法的号码
B#16#30: B#16#31: B#16#32: B#16#33: OB121_FLT_REG:	写访问至具有写保护的全局DB 写访问至具有写保护的背景DB 访问全局DB 时DB 号码故障 访问背景DB 时DB 号码故障 非法的DB 号
B#16#34: B#16#35: B#16#3A: B#16#3C: B#16#3D: B#16#3E: B#16#3F: OB121_FLT_REG:	在FC 调用时FC 号码故障 在FB 调用时FB 号码故障 访问未下装的DB, DB 号在允许范围 访问未下装的FC, FC 号在允许范围 访问未下装的SFC, SFC 号在允许范围 访问未下装的FB, FB 号在允许范围 访问未下装的SFB, SFB 号在允许范围 非法号码

## 2. 26 I/O 访问故障组织块 (OB122)

### 2. 26. 1 描述

当对模板的数据访问出现故障时CPU 的操作系统调用OB122。例如,如果CPU 对 I/O 模板的数据访问时检测到读故障, 操作系统调用OB122。

### 2. 26. 2 理解I/O 访问故障OB 的运行

OB122 与被中断的块在同一优先级中执行。如果OB122 未编程, CPU 从RUN 方式进入STOP 方式。S7 提供了下列SFC 在您的程序执行期间对OB122 进行屏蔽和解除屏蔽:

- SFC36 (MSK\_FLT) : 屏蔽特定的故障代码
- SFC37 (DMSK\_FLT) : 解除被SFC36 屏蔽的故障代码
- SFC38 (READ\_ERR) : 读故障寄存器

### 2. 26. 3 I/O 访问故障OB 的局部数据

下表描述I/O 访问故障OB 的临时（TEMP）变量。变量名为OB122 的缺省名。

变量	类型	描述
OB122_EV_CLASS	BYTE	事件级别和标识: B#16#29
OB122_SW_FLT	BYTE	故障代码: • B#16#42 对S7-300 和CPU 417: I/O 访问故障, 读操作对所有其它的S7-400CPU: 故障出现之后在第一次读访问期间故障• B#16#43 对S7-300 和CPU 417: I/O 访问故障, 写操作对所有其它的S7-400 CPU: 故障出现之后在第一次写访问期间故障 • B#16#44 (仅对S7-400, 不包括CPU 417) 在故障出现之后在第n 次 (n > 1) 读访问期间故障• B#16#45 (仅对S7-400, 不包括CPU 417) 在故障出现之后在第n 次 (n > 1) 写访问期间故障
OB122_PRIORITY	BYTE	优先级: 出现故障的OB 的优先级
OB122_OB_NUMBR	BYTE	OB 号 (122)
OB122_BLK_TYPE	BYTE	故障出现的块的类型 (B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB) (对S7-300 无有效值在这里记录)

变量	类型	描述
OB122_MEM_AREA	BYTE	存储器区和访问类型: • 位7 至4: 访问类型- 0: 位访问- 1: 字节访问- 2: 字访问- 3: 双字访问• 位3 至0: 存储器区- 0: I/O 区- 1: 过程映像输入- 2: 过程映像输出
OB122_MEM_ADDR	WORD	出现故障的存储器地址
OB122_BLK_NUM	WORD	引起故障的MC7 命令的块的号码 (S7-300 无效值在这里记录)
OB122_PRG_ADDR	WORD	引起故障的MC7 命令的相对地址 (S7-300 无效值在这里记录)
OB122_DATE_TIME	DATE_AND_TIME	OB 被调用时的日期和时间