SIEMENS

SIMATIC

# Process Control System PCS 7
# PCS 7 Basis Library V8.0

Function Manual




11/2011

A5E03710244-01

## Legal information

### Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

| ⚠DANGER |
|---|
| indicates that death or severe personal injury **will** result if proper precautions are not taken. |

| ⚠WARNING |
|---|
| indicates that death or severe personal injury **may** result if proper precautions are not taken. |

| ⚠CAUTION |
|---|
| with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken. |

| CAUTION |
|---|
| without a safety alert symbol, indicates that property damage can result if proper precautions are not taken. |

| NOTICE |
|---|
| indicates that an unintended result or situation can occur if the relevant information is not taken into account. |

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

### Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

### Proper use of Siemens products

Note the following:

| ⚠WARNING |
|---|
| Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed. |

### Trademarks



### Disclaimer of Liability



Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E03710244-01
Ⓟ 10/2011

# Table of contents

# 1 General Information About Block Description

The setup of the block description is always uniform and contains the following sections:

## Header of the block description

Example: CTRL_PID: PID controller block

The header begins with the type name of the block (e.g., “"CTRL_PID"). This symbol name is entered in the symbol table and must be unique within the project.
In addition to the type name, you will also see a keyword indicating the purpose or function of the block (e.g., "PID controller block").

## Object name (type + number)

FB x

The object name for the block type is made up of the type of implementation (function block = FB, function = FC) and the block number = x.

## Links for displaying block I/Os

Example:

- CTRL_PID block I/Os

Click the "Block I/Os" link to display a list of block I/Os for the designated block.

## Links for displaying the block icon and faceplate

If the block is intended for operator control and monitoring and a block icon and faceplate exist, the corresponding image and description can be displayed directly by clicking these links.

Example:

- CTRL_PID block icon
- CTRL_PID faceplate

## Function

Here, you will find a brief description of the block function.
You will find additional information about complex blocks in the "How it works" section.

## How it works

Here, you will find more detailed information, for example about the function of specific inputs, operating modes or time sequences. You must be familiar with these relationships in order to use the block effectively.

## Calling OBs

Here you will find information on the organization blocks (OBs), in which the described block must be installed. If the CFC is used, the block is automatically installed in the cyclic OB (cyclic interrupt) and in the OBs listed in the block's task list (for eample in restart OB100).

CFC generates the required OBs during compilation. If you use the blocks without CFC, you will have to program these OBs and call their instance within the blocks.

## Error handling

The **ENO** Boolean block output indicates the error in the CFC chart.
The value is equivalent to the **BIE** (binary result in STEP 7 STL, after completion of the block) or **OK** bit (in SCL notation) and indicates:

ENO = BIE = OK = 1 (TRUE) -> The result of the block is free of errors.

ENO = BIE = OK = 0 (FALSE) -> Invalid result or constraints (for example, input values and modes).

The FBs also return the inverted BIE at the **QERR** output of the instance DB.

QERR = NOT ENO

The error message is generated in two separate operations:

- The operating system detects a processing error (e.g. value overflow, system functions called return an error ID with BIE = 0).
  This is a system function and is not specifically mentioned in the block description.
- The block algorithm checks for functional invalidity of values and operating modes. These error events are logged in the block description.

You can evaluate the error display, for example, to generate messages or use substitute values for invalid results. You will find more information about messages in the "Message blocks" section.

## Startup characteristics

The different startup behaviors are as follows:

- Initial start

  The block is called for the first time from the OB in which it has been inserted. This is usually the OB that performs the standard, process-specific operations (for example, the cyclic interrupt OB).
  The block adopts a status that conforms to its input parameters. These may be default values (additional information in "I/Os" section) or values you have already configured, for example, in CFC. The initial startup characteristics are not described separately unless the block does not conform to this rule.

- Startup

  The block is executed once during CPU startup. The block is called in the startup OB (where it is additionally installed either automatically in the ES or manually in STEP 7). In this case, the startup characteristics are described.

  Please note that the block outputs have default values and that these can take effect during the CPU startup with other blocks, if these are processed first.

  The correct startup behavior of the blocks is the responsibility of the configuration engineer.

## Time response

A block assigned this function must be installed in a cyclic interrupt OB. It calculates its time constants/parameters on the basis of its sampling time (the time which elapses between two consecutive cyclic operations).

In a CFC configuration on ES, the sampling time is also determined by the segmentation of the runtime group, which ensures that the block is not executed during every OB run.

This sampling time is entered at the I/Os, in the SAMPLE_T parameter.

When configuring with CFC, this occurs automatically once the block has been inserted in the OB and the runtime group. For this reason, this input is set to invisible in CFC.

During the STEP 7 configuration, you set the time response manually.

Time response is mentioned only if the block has been assigned this feature.

## Message response

A block with message response reports various events to the higher level OS. Existing parameters required for the generation of messages are documented.

Blocks without message response can be expanded with additional message blocks. A reference to the message response is found in the description of the individual message blocks.

## I/Os

The I/Os of the block represent its data interface. These I/Os can be used either to transfer data to the block or to fetch results from the block.

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| U1 | Addend 1 | REAL | 0 |
| ..... | | | |

The "I/O" table lists all I/O parameters of the block type. You can access these lists using the engineering tools. They are in alphabetical order. Elements accessible only via the block algorithm (internal variables) are not listed.
The meaning of the columns is as follows:

- I/O

  Name of the parameter, derived from the English, e.g. PV_IN = **P**rocess **V**ariable **IN**put (process variable, controlled variable).
  The SIMATIC naming conventions have been applied.

  The block representation in CFC as supplied is as follows:
  I/O name in **bold** characters = I/O is visible, regular = I/O is invisible.

- Meaning

  Function (possibly also short description)

- Data type

  S7 data type of the parameter (BOOL, REAL, etc.)

- Default (default value)

  The value of the parameter before the block runs for the first time (unless changed in the configuration)

# Family: CONTROL 2

## 2.1 FM_CO: Coordination of function blocks which read data records

### 2.1.1 Description of FM_CO

#### Object name (type + number)

FB 79

- FM_CO block I/Os (Page 16)

#### Function

The block coordinates the data record reading of the blocks

| Block | Library | Module |
|---|---|---|
| FM_CNT | PCS 7 Basic Library | FM350-1, FM350-2 |
| FMCS_PID | PCS 7 Library | FM355 |
| FMT_PID | PCS 7 Library | FM355-2 |
| REAS355P | PCS 7 Library | FM355, FM355-2 |
| FmCont | PCS 7 APL | FM355 |
| FmTemp | PCS 7 APL | FM355-2 |

The block is installed and the parameters interconnected by the driver generator.

| Block | Configuration of the coordination number | Interconnections between FM_CO and the data record reading instance: FM_CO - DS reading instance |
|---|---|---|
| FM_CNT, FMCS_PID | CO_NO | EN_CO_x <-> EN_CO |
| FM_PID, READ355P | CO_NO | ENCOx_yy <-> ENCO |
| FmCont, FmTemp | CoordNo | EN_CO_x <-> EnCoord<br>ENCOx_yy <-> EnCoNum |

Where:

x = Sequence of data record reading blocks (0 to 7)

yy = Coordination number within a sequence (0 to 63)

It is possible to include data record reading blocks of other modules in the coordination.

## How it works

The FM_CO block can start a maximum of 8 block sequences.

The blocks connected to the output structure EN_COx check whether the current coordination number (EN_COx.CO_ACT) corresponds to their own coordination number (CO_NO/CoordNo). If this is the case, they read their data records from the module and reduce the coordination number EN_COx.CO_ACT by 1, so that the next block can read out its data records.

If the current coordination number of a sequence (EN_COx.CO_ACT) has a value less than 1, the FM_CO block determines the highest number assigned in sequence x based on its inputs ENCOx_yy. The data reading blocks supply the inputs ENCOx_yy with their respective coordination number via an interconnection. The highest coordination number is the number for which ENCOx_yy = yy still applies. The FM_CO module restarts the sequence in which it sets EN_COx.CO_ACT to this value.

This algorithm ensures that no more than one read data record operation ever takes place at any given time within the block sequence.

## Calling OBs

The fastest cyclic interrupt OB of all OBs in which you have installed data reading block instances and OB100 as well.

## Use in CFC

When using the CFC function "**Generate Module Drivers**", the block is automatically installed and the connections, such as those described under "Installation regulation" are made.

If you install, delete or move blocks of an existing block sequence in other OBs or runtime groups, the driver generator must be called.

Should the sequence not start up as expected (after CPU restart) or not continue to run (after downloading changes), you must set ACC_ID to 1.

## Installation rules/capacity

One FM_CO is responsible for one DP master system and has 8 sequences with data record reading block instances that are coordinated in parallel. The first block sequence contains instances of the data recording reading blocks that relate to the DP slaves 1, 9, 17 and so on. The second block sequence contains instances of the data recording reading blocks that relate to the DP slaves 2, 10, 18 and so on. The same principle applies to the remainder of the eight block sequences.

---

**Note**

When an instance is added to a sequence by a data recording reading block, the sampling time of all instances in the sequence increases. If, for example, an instance from DP slave 9 is added to the sequence 1, then the sampling time of all instances of sequence 1 increases including the instances from DP slave 1.

---

A sequence may contain up to 63 data recording reading instances of the following blocks:

| Block | Data record reading channels | Module |
|---|---|---|
| FM_CNT | One instance of FM_CNT is required for each module. 4 channels are read with one data record read operation per cycle | FM350-1, FM350-2 |
| FMCS_PID, FMT_PID, FmCont, FmTemp | The module has 4 controller channels. One channel is read with one data record read operation per cycle. | FM355, FM355-2 |
| READ355P | Certain process values of all 4 channels are read with one data record read operation per cycle. | FM355, FM355-2 |

The FM_CO must always be installed before the first data record reading instance in the fastest cyclic interrupt OB. The output structure EN_COx for the DP slave is connected to the input structures EN_COx or EnCoord of all data record reading instances that communicate with the controller modules of rack x. The output ENCO or EnCoNum of each data record reading instance is connected to an input ENCOx_yy (yy corresponds to the coordination number CO_NO or CoordNo assigned to each data recording reading instance) of the FM_CO block.

The selection of the cyclic interrupt OB depends on the CPU load. Note that the CPU has no reserves for other "Read data record" jobs if operating with eight or more DP slaves because only eight jobs can be buffered per DP master system. Simply inserting a module would lead to an overflow. It is advisable to operate only up to six DP slaves on a DP master system. The remaining DP slaves must be distributed on other DP master systems with further FM_CO blocks.

When selecting the cyclic OB, remember that the new data will be available at the earliest after two cycles. Make sure that the maximum runtime of this OB does not have any negative impact on overall system runtime as a result of the number of blocks installed. If the data recording reading block instances to be processed exceed the runtime limit, group the DP slaves with the FM350/FM355 modules in fast and slow control loops.

## Startup characteristics

EN_CO_x.CO_ACT = 1 is set at all outputs during startup (restart).

## Time response

Not available

## Message functionality

Not available

## 2.1.2 I/Os of FM_CO

The factory setting of the block display in CFC is identified in the "I/O" column: I/O name in **bold** characters = I/O is visible; I/O name in regular characters = I/O is invisible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ENCOx_yy | The coordination number yy is assigned in the rack x if the input has the value yy (x = 0 - 7, yy = 1 - 64) | BYTE | 0 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_CO_x | Coordination number of the block that can read data records | STRUCT | 0 |

### In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_ID | Restart all sequences | BOOL | 1 |

# Family: @System 3

## 3.1 CONEC: Monitoring the AS connection status

### 3.1.1 Description of CONEC

#### Object name (type + number)

FB 88

- CONEC Block I/Os (Page 20)

#### Area of application

The CONEC block monitors the status of AS connections, and reports the associated error events.

#### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 100 | Warm restart |

#### Use in CFC

With the "**Generate module drivers**" CFC function, the CONEC block is automatically installed in the OBs listed above.

#### Function and method of operation

The CONEC block generates messages which are output at ALARM_8P to WinCC (see "Message response"). For connection diagnostics, SFC 87 (C_DIAG) is called every 10 seconds in the cyclic interrupt OB (OB 32). Up to 128 connections are monitored.

---

**Note**

The messages "Failure or loss of redundancy connection ID" are generated by each CPU of the two connected AS except when the CPU of an AS fails (or both H-CPUs).
The connection ID determines whether a message is output.
If the connection ID >= 16#C00, **no** message is generated.

---

### Generation of the maintenance status MS

If any connection in the CONEC block is detected as having failed, the "Maintenance alarm" maintenance status is output.

If any connection in the CONEC block is detected as a redundancy loss, the "Maintenance request" maintenance status is output.

If in the CONEC block the messages are disabled via the parameter EN_MSG, then the maintenance status "Unchecked / Unknown" will be output.

### Error handling

Error handling for the block is limited to the evaluation of the error information of ALARM_8P. You will find more information in the

"Error Information of Output Parameter MSG_STAT" (Page 398) section.

### Startup characteristics

The CONEC block initializes the messages of ALARM_8P.

If there is a CPU with SFC 87, connection diagnostics is initialized. After this, there is a wait time of approx. 1 minute in the cyclic interrupt OB before the connection diagnostics messages are generated.

### Overload behavior

Not available

### Time response

For additional information, refer to "Message response".

### Message response

The block generates the following messages in the OBs listed below:

| OB | Start Event | Message |
|---|---|---|
| OB 32 | 1 sec. cyclic interrupt or alternative cyclic interrupt OB | Failure connection ID: xx incoming/outgoing<br>Loss of redundancy connection ID: xx entering/exiting state |

If EN_MSG = FALSE, messaging is disabled.

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of CONEC (Page 21)

Maintenance status of MS (Page 403)

## 3.1.2 I/Os of CONEC

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_MSG | 1 = Enable message | BOOL | 1 |
| EV_IDx | Message number for ALARM_8P_x<br>(x = 1 - 16, assigned by the ES) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| SAMPLE_T | Sampling time OB in seconds | REAL | 1.0 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| MSGSTATx | STATUS output of ALARM_8P_x (x = 1 - 16) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QMSGERx | Error output of ALARM_8P_x (x = 1 - 16) | BOOL | 0 |

### Additional information

For additional information, refer to the following sections:

Message texts and associated values of CONEC (Page 21)

Maintenance status of MS (Page 403)

### 3.1.3 Message texts and associated values of CONEC

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Default message text | Message class (Page 401) |
|---|---|---|---|
| EV_ID1 to EV_ID8 | 1 | Failure connection ID: 16#@1%X@ | S |
| | 2 | Failure connection ID: 16#@2%X@ | S |
| | 3 | Failure connection ID: 16#@3%X@ | S |
| | 4 | Failure connection ID: 16#@4%X@ | S |
| | 5 | Failure connection ID: 16#@5%X@ | S |
| | 6 | Failure connection ID: 16#@6%X@ | S |
| | 7 | Failure connection ID: 16#@7%X@ | S |
| | 8 | Failure connection ID: 16#@8%X@ | S |
| EV_ID9 to EV_ID16 | 1 | Loss of redundancy connection ID: 16#@1%X@ | F |
| | 2 | Loss of redundancy connection ID: 16#@2%X@ | F |
| | 3 | Loss of redundancy connection ID: 16#@3%X@ | F |
| | 4 | Loss of redundancy connection ID: 16#@4%X@ | F |
| | 5 | Loss of redundancy connection ID: 16#@5%X@ | F |
| | 6 | Loss of redundancy connection ID: 16#@6%X@ | F |
| | 7 | Loss of redundancy connection ID: 16#@7%X@ | F |
| | 8 | Loss of redundancy connection ID: 16#@8%X@ | F |

## Assignment of associated values

The process control messages are generated by ALARM_8P with EV_ID1 to EV_ID16 with 8 associated values. The table below shows how the associated values are assigned to the block parameters.

| Message block ALARM_8P | Associated value | Block parameter | Data type |
|---|---|---|---|
| EV_ID1... EV_ID16 | 1 | Connection_ID 1+x | WORD |
| | 2 | Connection_ID 2+x | WORD |
| | 3 | Connection_ID 3+x | WORD |
| | 4 | Connection_ID 4+x | WORD |
| | 5 | Connection_ID 5+x | WORD |
| | 6 | Connection_ID 6+x | WORD |
| | 7 | Connection_ID 7+x | WORD |
| | 8 | Connection_ID 8+x | WORD |

x = 0 for EV_ID1, x = 8 for EV_ID2, x = 16 for EV_ID3 etc. to x = 56 for EV_ID8
x = 0 for EV_ID9, x = 8 for EV_ID10, x = 16 for EV_ID11 etc. to x = 56 for EV_ID16

# 3.2 CPU_RT: Determination of the runtime of OBs

## 3.2.1 Description of CPU_RT

### Object name (type + number)

FB 128

- CPU_RT block I/Os (Page 29)

### Area of application

The CPU_RT block is installed by the CFC in OB 100, OB 1, in all OB 3x as well as OB 8x, if this is used by the user program.

The CPU_RT determines the runtime of the individual OBs and their participation in the cycle time. If there is CPU overload (OB 80 cycle time exceeded), it instigates suitable actions selected by the user in limits to ensure operability of the AS.

This situation is designated as emergency operation and is made clearly visible by a process control message. Buffered start events (OB 3x still executing) are also detected and displayed. The loss of start events is reported as error.

### Use in CFC

During compilation of the CFC, a chart is automatically created with the name @CPU_RT. The CPU_RT block is already included in it.

**Note**

Never attempt to insert the CPU_RT block in a different block because it is a system block.

## Function and method of operation

At CPU restart and when downloading changes the slowest OB 3x is determined with SZL ID 822 (data records of all assigned alarms of an alarm class).

Note: The slowest cyclic OB 3x (slowest OB) must also have the lowest priority set so that a useful analysis is possible.

In OB-BEGIN, if there are implausible settings, a message EV_ID2 signal 3 "Priorities of cyclic OBs do not conform to PCS 7" is output and the maintenance status (MS is set to "Maintenance demand" = 16#00000005).

SFC78 is used to determine the OB runtimes. If it is not present, no warning limit will be output as a message.

---

**Note**

Older CPUs do not support SFC78. Use SSL112 to check whether SFC78 is available.

---

**Note**

The status of CPU_RT is reset when you download.

---

## Behavior at higher CPU load

If the average value of all net runtimes (in % of OB 3x, OB 8x + OB 1) exceeds the value MAX_LIM, then in OB_BEGIN, the message EV_ID2- signal 1 "Net time consumption of all OBs exceeds max limit" is output.

The maintenance status MS is set to "Maintenance demand" = 16#00000005 in OB_BEGIN.

The message and MS are cleared with a value less than MAX_LIM – HYS.

## Behavior in the event of OB request errors

If a programmable number of these OB 3x events has been exceeded or if an OB 1 event is detected without an OB 1 having been processed, the message EV_ID3 – Signal 2 "OB request: OB 3x still being processed" is output in OB_BEGIN. The number of OB 3x events can be set at the input "N_REG_ERR"; the default value = 4.

The maintenance status MS is set to "Uncertain maintenance demanded" = 16#00000006 in OB_BEGIN.

If an OB 1 is then run through again, the MS is reset and this process control message will be marked as "outgoing".

In the faceplate of OB_BEGIN, there is a display of the request error for each OB 3x. The first occurrence of a request error is be displayed. These displays can be reset with the reset key.

## Behavior when the maximum cycle time is exceeded

If the maximum cycle time is exceeded, a message EV_ID3 – Signal 1 "Cycle time exceeded: @1d@ OB@2d@" is output.

The maintenance status MS is set to "Bad or maintenance alarm" = 16#00000007 in OB_BEGIN.

If an OB 1 is then run through again, the MS is reset and this process control message will be marked as "outgoing".

## Behavior to prevent stop

If the cycle time is exceeded twice without an OB 1 being processed, this results in Emergency Operation with stop avoidance activated. The process control message EV_ID1 – Signal 3 "Emergency operation, cycl. OBs will be reduced" will be output.

The maintenance status MS is set to "Bad or maintenance alarm" = 16#00000007.

If the CPU resumes normal operation after the problem has been eliminated, the MS is reset and this process control message is marked as "outgoing".

## Behavior during downloading

The status of CPU_RT is reset when you download.

## Measures for avoiding stop

When the CPU is overloaded you can prevent the CPU from becoming inoperable by "load shedding". Load shedding is achieved by suspending the cyclic levels and is an emergency mode. The user can still exclude individual OBs for the first escalation stage, for example the level with the F drivers.

To avoid a CPU stop, CPU_RT takes the following measures in OB 80 when reaching a cycle overflow occurs:

- Cycle time monitoring is triggered with an SFC43 call, to prevent a CPU stop.
- A memory bit is set to detect the next immediate OB 80 call within an OB 1 call, so that, if necessary, measures can be initiated that prevent an overload of the AS.

In OB 80, the measures to prevent the overload are initiated and they reversed in the slowest OB.

Two escalation stages can be set:

1. Stage: None of the used OB 3x blocks will be processed for one cycle, unless they have been excluded by the user (OB3x_ATTN = FALSE).

2. Stage: Now all previously excluded OB 3x blocks will likewise not be processed for one cycle. If this does not have a steadying effect, whenever OB 3x blocks are executed their execution will be suspended again for one cycle.

Assign the parameters in CPU_RT for each OB 3x at the following inputs:

| | |
|---|---|
| OB3x_ATTN = TRUE | The OB is included in the measures to prevent overload. Default is "TRUE". |

The maximum number of SFC43 calls can be set at the input MAX_RTRG. If the maximum number x is exceeded, the CPU goes to stop.

The number x is reset when there is an OB 1 call again.

If you set MAX_RTRG = 0, then the function stop avoidance on overload function is deactivated.

If the measures are effective, in other words OB 1 is run through again, a calculation is made at that point to determine whether canceling the measures would again result in overload. If yes, the measures remain in effect. The measures are reduced step-by-step, when safe operation is possible again.

## Reversal of the measures for stop avoidance

To initiate a reversal the percentage sum of the cyclic OBs calculated back to a lower reduction ratio, must be less than full CPU utilization.

Use the parameter MAX_VAL to set the value that corresponds to full CPU utilization. The value "95" is default.

The calculation is made according to the following formula:

((NET30PERint * (OB30_N_START+1) / OB30_N_START)+
 (NET31PERint * (OB31_N_START+1) / OB31_N_START)+
 (NET32PERint * (OB32_N_START+1) / OB32_N_START)+
 (NET33PERint * (OB33_N_START+1) / OB33_N_START)+
 (NET34PERint * (OB34_N_START+1) / OB34_N_START)+
 (NET35PERint * (OB35_N_START+1) / OB35_N_START)+
 (NET36PERint * (OB36_N_START+1) / OB36_N_START)+
 (NET37PERint * (OB37_N_START+1) / OB37_N_START)+
 (NET38PERint * (OB38_N_START+1) / OB38_N_START)+
 NET01PER)< MAX_VAL

NETxxPERint is the percentage share of a cyclic OB in the total runtime as a mean value and
(OB30_N_START+1) is the current reduction factor of the OB.

The net percentage values are also mean values, because in case of reduction, averaging is a must.

For the calculation, a separate mean value generation was used that has a separate sample factor (SAMPLE_RE).

If the condition is satisfied, then after a number of cycles in the slowest OB (parameter "UndoCycle") the reduction factor is decremented by 1 for all OBs.

If the total sum of the OBs is still below MAX_VAL after this, then after a number of cycles in the slowest OB (UndoCycle), the factor will continue to be decremented until the used cyclic OBs have reached the reduction factor 1.

After this, for the OBs (OB3x_ATTN = FALSE) excluded by the user, the reduction ratio will be set to 0.

Finally. the reduction ratio will be set to 0 for all other cyclic OBs.

If no SFC78 is present, then the time at which reversal of the stop avoidance measures can be triggered cannot be calculated.

The reversal of reduction ratios is started when the slowest OB has again processed a number of cycles (UndoCycle).

The value of the UndoCycle in this case should not be too low, to avoid a frequent back and forth between stop avoidance measures and normal operation.

For the reduction ratio in the CFC, two parameters are available in the CPU_RT block for each cyclic OB:

| | |
|---|---|
| OB3x_N_START | The start value for reduction ratio is specified by the input OB3x_N of CPU_RT and also in OB3x_N_CNT |
| OB3x_N_CNT | The counter is decremented in the CFC at each OB call. For OB3x_N_CNT <= 0 there is complete OB processing and OB3x_N_START will be re-entered in OB3x_N_CNT. |

The CPU block is also called when reduction becomes necessary, so that in emergency operation an evaluation of the averaged cycle time is possible.

## Utilization display with SFC78

When the block executes, the calling OB is determined. For every OB with the system function SFC78, it reads the net runtime, LAST_RTxx, and the gross runtime LAST_Etxx of the last completed OB processing. The absolute times are specified in milliseconds.

The mean value is generated for each OB (OB 3x and OB 8x) according to the formula

Mean value = mean value + (new value – mean value ) / sample_AV

SAMPLE_AV is a parameter that is calculated separately for each OB 3x, OB 8x. If necessary, you can adapt the precision of the mean value with this parameter that is based on the number of measuring cycles, or on the influence of the new value on the mean value.

The default of SAMPLE_AV = 25 cycles, based on the slowest cyclic OB.

The actual sample factor for specific OBs is determined according to the following formula:

SampleOB3x = Sample * execution frequency of the slowest OB / execution frequency of OB3x

This ensures equally-weighted mean value generation over time for all cyclic OBs.

The sample factor for the total mean value of OB 1, and for OB 8x, is set to = sample, because the basis for these values is also the execution frequency of the slowest OB.

Use the reset button on the faceplate of OB_BEGIN to reset mean value generation.

After the reset, the divisor "Sample" of 1 per cycle is incremented by 1.

This means that with the reset, mean value generation is initially imprecise and it becomes more precise through the factor "sample". The influence of the new value on the mean value initially is significant and diminishes as the "sample" factor becomes greater.

By resetting the data, a fresh measuring cycle can be started at anytime.

The reset of all mean values takes place in OB 1. Mean value generation does not take place during this time.

### Error handling

If the read-out of data from the cyclic OB fails for the CPU_RT block, then ERR_NUM = 1 is set and processing of the CPU_RT block is abandoned, because these data are the basic prerequisite for useful processing.

### Startup characteristics

Calculations with SFC78 are restarted only after a number of cycles (RunUpCyc) after restart. The RunUpCyles are counted down in the slowest cyclic OB.

### Time response

Not applicable.

### Message response

The block reports via OB_BEGIN (Page 217)

### Operator control and monitoring:

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostics screens have been generated, the faceplate can be called via the block icon of the AS.

- OB_BEGIN faceplate
- Asset Management block icons (for more information, refer to the "Process Control System PCS 7; Maintenance Station" manual)

If no asset management is used in the project, the "OB-BEGIN" block icon is used to display avoidance of stop.

### Additional information

You will find more information on this subject in the following sections:

Message texts and associated values of OB_BEGIN (Page 224)

Maintenance status of MS (Page 403)

### 3.2.2 I/Os of CPU_RT

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O | Meaning | Type | Default |
|---|---|---|---|
| DELTA_L | Flag for change compile | BOOL | 1 |
| **HYS** | Hysteresis of the max. total number | INT | 5 |
| **MAX_LIM** | Max. total number | REAL | 75 |
| **MAX_RTRG** | Max. number of calls (for SFC 43) | INT | 50 |
| **MAX_VAL** | Max. value for calculating the reset of reduction ratios | REAL | 95 |
| **N_REQ_ERR** | Number of OB 3x request errors | INT | 4 |
| **OB3x_ATTN** | (x = 0 – 8) OB 3x: 1 = participates in measures to prevent overloads | BOOL | 1 |
| **RESET** | Resets the mean values, minimum values, and maximum values | BOOL | 1 |
| RUNUPCYC | Number of start-up cycles | INT | 5 |
| SAMPLE_AV | Sample factor for mean value generation | INT | 50 |
| SAMPLE_RE | Sample factor for mean value generation internal | INT | 50 |
| **UNDO_CYC** | Counter in the slowest OB for emergency operation | INT | 100 |

#### Output parameters

| I/O | Meaning | Type | Default |
|---|---|---|---|
| **CPU_RT_DATA** | System structure: Performance data | STRUCT | |
| **DAT_PLAU** | 1 = slowest OB 3x has the lowest priority | BOOL | 0 |
| ERR_NUM | 1 = occurrence of an error | INT | |
| EXC_FR3x | (x = 0 – 8) execution cycle (in ms) of the OB 3x | INT | 0 |
| GRO3xAV | (x = 0 – 8) gross mean value | REAL | 0 |
| **GRO3xCUR** | (x = 0 – 8) gross current value | REAL | 0 |
| GRO3xMAX | (x = 0 – 8) gross maximum value | REAL | 0 |
| GRO3xMIN | (x = 0 – 8) gross minimum value | REAL | 0 |
| GRO3xPER | (x = 0 – 8) gross mean value (in %) | REAL | 0 |
| **MAXCYCTI** | Set scan cycle monitoring time | INT | 0 |
| **N_OB1_CYC** | Number of OB 1 calls during a cycle of the slowest OB | INT | 0 |
| NET01AV | Net mean value of OB 1 (in ms) | REAL | 0 |
| **NET01CUR** | Net current value of OB 1 (in ms) | REAL | 0 |
| NET01MAX | Net maximum value of OB 1 (in ms) | REAL | 0 |

| I/O | Meaning | Type | Default |
|---|---|---|---|
| NET01MIN | Net minimum value of OB 1 (in ms) | REAL | 0 |
| NET01PER | Net mean value of OB 1 (in %) | REAL | 0 |
| NET3xAV | (x = 0 – 8) net mean value of OB 3x (in ms) | REAL | 0 |
| NET3xCUR | (x = 0 – 8) net current value of OB 3x (in ms) | REAL | 0 |
| NET3xMAX | (x = 0 – 8) net maximum value of OB 3x (in ms) | REAL | 0 |
| NET3xMIN | (x = 0 – 8) net minimum value of OB 3x (in ms) | DINT | 0 |
| NET3xPER | (x = 0 – 8) net mean value of OB 3x (in %) | REAL | 0 |
| NET8xAV | (x = 0 – 8) net mean value of OB 8x (in ms) | REAL | 0 |
| NET8xCUR | (x = 0 – 8) net current value of OB 8x (in ms) | REAL | 0 |
| NET8xMAX | (x = 0 – 8) net maximum value of OB 8x (in ms) | REAL | 0 |
| NET8xPER | (x = 0 – 8) net mean value of OB 8x (in %) | REAL | 0 |
| OB3x_N_CNT | (x = 0 – 8) decrementing counter for reduction ratio | INT | 0 |
| OB3x_N_START | (x = 0 – 8) start value for reduction ratio | INT | 0 |
| REQ01ERR | (x = 0 – 8) OB request errors since the last reset | BOOL | 0 |
| REQ3xERR | (x = 0 – 8) OB request errors | BOOL | 0 |
| SFC78_EX | 1 = SFC 78 available in CPU | BOOL | 0 |
| SL_OB | Slowest OB 3x | BYTE | 0 |
| SL_OB_EXC_FR | Number of calls of the slowest OB 3x | INT | 0 |
| TOTALAV | Total average value of all OB 1, OB 3x, OB 8x (in %) | DINT | 0 |
| TOTALCUR | Total current value of all OB 1, OB 3x, OB 8x (in %) | DINT | 0 |
| TOTALMAX | Total maximum value of all OB 1, OB 3x, OB 8x (in %) | DINT | 0 |
| TOTALMIN | Total minimum value of all OB 1, OB 3x, OB 8x (in %) | DINT | 0 |

?

| I/O | Meaning | Type | Default |
|---|---|---|---|
| IDLE_CYC | CPU utilization display | INT | 0 |

## Additional information

You will find more information on this subject in the following sections:

Message texts and associated values of OB_BEGIN (Page 224)

# 3.3 DIAG_AB: Evaluation of statusword AB7000

## 3.3.1 Description of DIAG_OB

### Object name (type + number)

FB 414

- DIAG_AB Block I/Os (Page 33)

### Area of application

The DIAG_AB block evaluates the status word of an AB7000 slave and acknowledges newly reported errors via the control word of the slave.

### Calling OBs

The cyclic OB and OB 100.

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The block is installed in the run sequence before the MOD_PAL0 or MOD_PAX0 block, both of which are also installed by the driver generator. The install is executed in the same cyclic OB as the associated signal processing blocks FF_A_xx.
- Parameters are assigned to the LADDR_C input with the address of the control word of the AB7000.
- Parameters are assigned to the input LADDR_S with the address control word of the AB7000.
- The OUT structure CPU_DIAG of the OB_BEGIN block is interconnected with the IN_OUT structures of the same name of DIAG_AB.
- The input mode of the DIAG_AB block is interconnected with the output OMODE_00 of the PADP_L10 or PADP_L01 block.
- The input PA_DIAG of the DIAG_AB block is interconnected with the output PA_DIAG of the PADP_L10 or PADP_L01 block.
- The output OMODE of the DIAG_AB block is interconnected with the input MODE_00 of the MOD_PAL0 or MOD_PAX0 block.
- The output ODIAG of the DIAG_AB block is interconnected with the input PA_DIAG of the MOD_PAL0 or MOD_PAX0 block.

## Function and method of operation

The DIAG_AB block cyclically analyses the status word of the AB7000 slave,

If a Modbus device fails, or if there is a higher-level error at the MODE input, then the OMODE and PA_DIAG outputs are set to "Bad":

| Parameters | Value | Description |
|---|---|---|
| OMODE | 16#40000001 | Higher level error |
| ODIAG | 16#00400000 | Due to process no valid values |

After an error exiting state the outputs are set to the status "Good":

| Parameters | Value | Description |
|---|---|---|
| OMODE | 16#80000001 | Valid value |
| ODIAG | PA_DIAG | Diagnostics information from PADP_L10 or PADP_L01 block |

The outputs SR_CODE and SR_DATA show the last values of a status tab sent by the AB7000. The meaning of SR_DATA depends on SR_CODE:

| SR_CODE | SR_DATA | Description |
|---|---|---|
| 16#00 | Number of re-transmissions | Reading or writing an FIM tab needed to be executed again due to an error |
| 16#01 | Address of the FIM | No connection to the FIM |
| 16#03 | Address of the FIM | The FIM has sent more data than expected |
| 16#04 | Address of the FIM | An error has occurred, no more data is available |
| 16#13 | | No error if SR_DATA = 16#00; otherwise, failure of the FIM (with the address in SR_DATA) |
| 16#1F | --- | An error is no longer present |

## Error handling

The validity of input parameters is not checked.

## Startup characteristics

Initialization of outputs OMODE with 16#80000001 ("valid value") and ODIAG with 16#00000000 ("no error")

## Time response

Not available

## Message response

Not available

## Operator control and monitoring

Not available

## 3.3.2 I/Os of DIAG_AB

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: General Information About Block Description (Page 9).

### Input parameters

| I/O | Meaning | Type | Default |
|---|---|---|---|
| **LADDR_C** | Logical address of the control word | INT | 0 |
| **LADDR_S** | Logical address of the control word | INT | 0 |
| **MODE** | Value status | DWORD | 16#80 000 000 |
| **PA_DIAG** | Diagnostic information | DWORD | 0 |

### Output parameters

| I/O | Meaning | Type | Default |
|---|---|---|---|
| **ODIAG** | Field devices diagnostics information | DWORD | 0 |
| **OMODE** | Value status of the slave | DWORD | 0 |
| **SR_CODE** | Code of the status tab | BYTE | 0 |
| **SR_DATA** | Data of the status tab | BYTE | 0 |

### In-out parameters

| I/O | Meaning | Type | Default |
|---|---|---|---|
| **CPU_DIAG** | CPU diagnostics (system structure) | STRUCT | |

# 3.4 DPAY_V0: Monitoring DP/PA and Y-Link operating as V0 slave

## 3.4.1 Description of DPAY_V0

### Object name (type + number)

FB 108

- DPAY_V0 Block I/Os (Page 39)

### Area of application

Block DPAY_V0 monitors the status of a DP/PA or Y-Link as a V0 slave (IM 157) and reports the corresponding error events.

The DP/PA link operates as a PA master for the lower-level PA field devices and as a slave on the DP bus.

The Y-Link operates as a DP master for the lower-level DP field devices and as a slave on the higher-level DP bus.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 70 | I/O redundancy error |
| OB 72 | CPU redundancy error |
| OB 82 | Diagnostic interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The block is integrated in the run sequence downstream from the SUBNET block and upstream from the PADP_L0x block.
- RACK_NO (rack/station number) is configured.
- SUBN_TYP (internal/external PROFIBUS interface) is set.
- SUBN1_ID (ID of the master systems) is set.
- SUBN2_ID (ID of the redundant master system) is set.
- DADDR (diagnostic address of the DP/PA or Y-Link) is set.
- DPPA_xx (slave xx address), 1st module (slot) address of slave xx in the link, number of slots of slave xx are set.
- The CPU_DIAG of the OB_BEGIN block and SUB_DIAG of the SUBNET block OUT structures are interconnected with the IN_OUT structures of the same name of DPAY_V0.
- In the case of PA or DP field devices, they are interconnected with PADP_L0x.

## Function and method of operation

If redundancy losses and link failures occur, the DPAY_V0 block generates a control-system error message for the OS. The block also indicates error events at active links (SUBN1ERR, SUBN2ERR) and at the preferred channel (SUBN1ACT, SUBN2ACT) in the output status bar. The output structure RAC_DIAG contains the geographic address of the link as well as the group error information RACK_ERR. The corresponding link is not available if RACK_ERR = 1.

The block requires a PROFIBUS DP interface. This can either be integrated in the CPU or provided by means of an external DP interface (CP). PROFIBUS DP is converted to PROFIBUS PA by means of a SIMATIC DP/PA-Link.

The field devices of a link are always addressed at the higher-level DP bus via the DP address of IM 157.

The AS addresses the field devices via the link, i.e., indirectly. The topological structure of the PA bus is mapped in the flat structure of the slave interface. A maximum of 64 field devices can be operated downstream from a link. Each field device can use any number of virtual slots at the link, up to 223 maximum.

In order to enable the assignment of diagnostic data to the field devices, the block provides each field device a DPPA_xx input structure consisting of 3 bytes with the following contents:

- Byte (SLAV_NO) = node number (address) of the field device at the PA/DP master system of the LINK
- Byte (SLOT_NO) = 1st module address of the field device in the link
- Byte (SLAV_SL) = number of slots of the field device

The "Generate module drivers" CFC function fetches this data from HW Config.

The start information is read from the CPU_DIAG I/O structure. This structure must be interconnected to the CPU_DIAG structure of the OB_BEGIN block (carried out by the CFC function "Generate module drivers").

The block generates a corresponding message (see "Message Response") on the basis of the startup information of calling OBs, if the current instance is affected.

When operating with redundant PROFIBUS DP interfaces, the block determines the currently active preferred channel (SUBN1ACT, SUBN2ACT) by evaluating the error events as well as via the diagnostic address DADDR of the link.

SFC 13 (DPNRM_DG, read diagnostic data consistently) reads the diagnostic data (OB 82). The reading process can take several cycles (OB 1). It is therefore possible in a few rare cases that the triggering diagnostic event cannot be recognized.

Diagnostic user data contains information about the status of the link, and of connected field devices. The structure DPPA_ST indicates the link status.

The status of a field device is entered in the structure DPA_M_xx.
A field device can have a maximum of 32 slots (modules). Three block types are available, according to the number of slots on the field device:

- PADP_L00 (field device with max. 7 slots)
- PADP_L01 (field device with max. 16 slots)
- PADP_L02 (field device with max. 32 slots)

The structure DPA_M_xx is interconnected to the structure DPA_M and the output EN_Mx with EN of one of the PADP_Lxx blocks (carried out by the CFC function "Generate module drivers").

The DPA_M_xx structure consists of two DWORD value (S_01 for modules 1 to 16 and S_02 for modules 17 to 32) and one BOOL value (S_ERR = DP/PA field device faulty). Two bits of the DWORD are assigned to each slot of the DP/PA field device, whereby bit 0 and bit 1 belong to slot 1 (module 1) of the DP/PA field device, etc. These bits are defined as follows:

| Status Bit 0 | Status Bit 1 | Meaning |
|---|---|---|
| 0 | 0 | Module x OK (valid user data) |
| 0 | 1 | Module x error (invalid user data) |
| 1 | 0 | Wrong module x (invalid user data) |
| 1 | 1 | No module x (invalid user data) |

If the diagnostics alarm applies to the entire DP/PA field device, then DPA_M_xx.S_ERR = TRUE is set.

Note: If you want to change the SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs online, you must set input ACC_ID = TRUE. This verifies the Link states and updates output values.

## Redundancy

The block supports redundant DP master systems in an H system (distributed I/Os only). The SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs of the SUBNET block are configured with the numbers of the redundant DP master systems. If the DP master systems are not redundant, the remaining input is set to 16#FF (default).

## Error handling

Error handling for the block is limited to the evaluation of the error information of ALARM_8P. For more information on this, refer to the section:

Error information of the MSG_STAT output parameter (Page 398)" section.

## Startup characteristics

The block initializes the messages of ALARM_8P. Availability of the link is verified. In H systems, determines the preferred channel of the link.

## Overload behavior

The block counts OB 86 (no DP master system failure, see SUBNET block) and OB 82 calls. Both counters are reset in OB 1. If more than five OB 86 events or more than five OB 82 events in succession before the cycle control point is reached (OB 1), these events are discarded and the message ""DP-Link DP-Master:x Rack:y: Multiple failure" or the message "DP-Link Master:x Rack:y: Muktiple alarm (OB 82)" is output. 1 minute later the status of the link will be re-checked.

## Time response

Not available

## Message response

After its call by OB 70, OB 72, OB 85 or OB 86, the block analyzes the status of its assigned CPU, DP master and link. If the link loses redundancy or fails, the block outputs corresponding messages via ALARM_8P.

The block generally reports only the events generated in the link that it monitors. Redundancy loss and link failures which are caused by the failure of a DP masters or of a CPU, are initially neither signaled nor indicated at the outputs SUBN1ERR and SUBN2ERR.

The DELAY input is used to delay the output of error messages for higher-priority outgoing errors. This delay time is configurable. When the block recognizes an outgoing error at an interconnected DP master, it initially assumes that there is a faulty assigned DP slave in the link it monitors and sets the corresponding output SUBNxERR. The error status is not reset until the DP slave returns (in this case: OB 86, OB 70). The blocks delay error messages relevant to any slave failure states for a time in seconds as specified in DELAY, in order not to trigger the output of surge of messages from DP slaves which are not yet synchronized after the master has returned. An error message is only output to the OS when the DP slave has not reported its return before this delay time has expired.

Do not set the value of DELAY too high, since messages reporting faulty DP slaves or their removal during a master failure will be output too late to the OS after the DP master returns.

The block generates the following messages in the OBs listed below:

| OB no. | Start Event | Message |
|---|---|---|
| OB 1 | Cyclic processing | Repeat the update of ALARM_8P outputs/messages, if necessary |
| OB 70 | Redundancy loss | Link redundancy loss/return |
| OB 85 | Program execution error | Link failure going |
| OB 86 | Rack failure | Link failure coming/going |
| OB 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the sections:

Message texts and associated values of DPAY_V0 (Page 41)

Maintenance status of MS (Page 403)

## 3.4.2 I/Os of DPAY_V0

### I/Os

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: General Information About Block Description (Page 9).

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **DADDR** | Diagnostic address of the DP slave | INT | 0 |
| DELAY | Interrupt delay (s) | INT | 15 |
| DPPA_xx | Information of the DP/PA slave (xx = 00 - 63) | STRUCT | |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack/station number | WORD | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DPA_M_xx | Status of the DP/PA slave (xx = 00 - 63) | STRUCT | |
| **DPPA_ST** | DP/PA/Y-Link status | STRUCT | |
| EN_Mxx | 1 = Enable modules (xx = 00 - 63) | BOOL | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| **RAC_DIAG** | Rack diagnostics | BOOL | 0 |
| **SUBN1ACT** | 1 = Slave 1 is active | BOOL | 0 |
| **SUBN1ERR** | 1 = Error in DP master system 1 | BOOL | 0 |
| **SUBN2ACT** | 1 = Slave 2 is active | BOOL | 0 |
| **SUBN2ERR** | 1 = Error in DP master system 2 | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | CPU diagnostics (system structure) | STRUCT | |
| SUB_DIAG | OB startup information | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of DPAY_V0 (Page 41)

Maintenance status of MS (Page 403)

## 3.4.3 Message texts and associated values of DPAY_V0

### Assignment of message text and message class

| Message no. | Default message text | Message class |
|---|---|---|
| 1 | DP Link @1%d@/ @3%d@: Redundancy loss | S |
| 2 | DP Link @2%d@/ @3%d@: Redundancy loss | S |
| 3 | DP Link @1%d@/ @3%d@: Failure | S |
| 4 | DP Link @2%d@/ @3%d@: Failure | S |
| 5 | - | - |
| 6 | - | - |
| 7 | DP LINK @1%d@/ @3%d@: Multiple alarm (OB 82) | S |
| 8 | DP LINK @1%d@/ @3%d@: Multiple failure | S |

### Assignment of associated values

| Associated value | Block parameter |
|---|---|
| 1 | ID of the primary DP master system (SUBN1_ID) |
| 2 | ID of the redundant DP master system (SUBN2_ID) |
| 3 | Rack/station number (RACK_NO) |

### See also

Message Classes (Page 401)

# 3.5 DPAY_V1: Enabling blocks downstream of a DP/PA and Y-Link operating as V1 slave

## 3.5.1 Description of DPAY_V1

### Object name (type + number)

FB 115

- DPAY_V1 Block I/Os (Page 45)

### Area of application

The DPAY_V1 block enables the field device-specific blocks downstream of the DP/PA or Y links.

The DP/PA link acts as a PA master for the lower-level PA field devices, and as a slave on the DP bus.

The Y link acts as a DP master for the lower-level DP field devices, and as a slave on the higher-level DP bus.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 55 | Status interrupt |
| OB 56 | Update interrupt |
| OB 57 | Vendor-specific interrupts |
| OB 70 | I/O redundancy error |
| OB 72 | CPU redundancy error |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The block is integrated in the run sequence after the OB_DIAG1 block.
- SUBN_1ID (ID of primary DP master system) is configured.
- SUBN_2ID (ID of secondary DP master system) is configured.
- RACK_NO (rack/station number) is configured.
- The OUT structure CPU_DIAG of the OB_BEGIN block is interconnected with the IN_OUT structures of the same name of DPAY_V1.
- The OUT structure CPU_OB_5X of the OB_BEGIN block is interconnected with the IN_OUT structures of the same name of DPAY_V1.
- EN_Mxx are interconnected with EN of OB_DIAG1 and PADP_L10 for each field device.

## Function and method of operation

The start information is read from the CPU_DIAG I/O structure. The structure must be interconnected with the CPU_DIAG structure of OB_BEGIN (carried out by the CFC function “Generate module drivers”). The affected downstream blocks will be enabled according to the start information.

## OB 5x characteristics

Enables the output for the affected field device.

## Redundancy

The redundancy is evaluated in OB_DIAG1.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

The block initializes its outputs.

## Overload behavior

OB_DIAG1 disables the block in response to an overload.

## Time response

Not available

### Message response

Not available

### Operator control and monitoring

The block has no faceplate.

### 3.5.2 I/Os of DPAY V1

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **CPU_DIAG** | CPU diagnostics (system structure) | STRUCT | |
| **CPU_OB_5X** | OB_5x startup information | STRUCT | |
| DPPA_xx | Information about the DP/PA slave (xx = 00 - 63) | STRUCT | |
| **RACK_NO** | Rack/station number | WORD | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

#### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **CPU_DIAG** | CPU diagnostics (system structure) | STRUCT | |
| **CPU_OB_5X** | OB_5x startup information | STRUCT | |
| EN_Mxx | 1 = Enable slave (xx = 00 - 63) | BOOL | 0 |

## 3.6 DPDIAGV0: Monitoring the status of ET 200S modules acting as DPV0 slaves after the Y link

### 3.6.1 Description of DPDIAGV0

#### Object name (type + number)

FB 117

- DPDIAGV0 Block I/Os (Page 49)

#### Area of application

The DPDIAGV0 block monitors the status of the modules of an ET 200S acting as a DPV0 slave (IM 151-1 High Feature) after a Y link.

#### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The block is integrated in the run sequence after the OB_DIAG1 block.
- The following inputs are configured:
  - SUBN_1ID (ID primary DP master system)
  - SUBN_2ID (ID secondary DP master system)
  - RACK_NO (rack/station number)
- The following I/Os are interconnected:
  - The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block with the DPDIAGV0-block IN_OUT structures of the same name
  - EN_Mxx with EN of the OB_DIAG1 block and the DPDIAGV0 block of each ET 200S
  - The DPA_M_xx outputs with the DPA_M input and EN_Mxx output with EN of a MOD_4 block.

## Function and method of operation

In the event of a diagnostic interrupt, the DPDIAGV0 block analyzes the ID-specific diagnostic data, and the module status of an ET 200S in DPV0 mode after a Y link.

The upstream OB_DIAG1 block detects the failure/restart of an ET 200S.

The AS addresses the devices via the link, i.e., indirectly. The topological structure of the DP bus is mapped in the flat structure of the slave interface. There may be up to 64 devices downstream of a Link. Each device can be assigned any number of virtual slots (max. 223) of the link. To assign the diagnostic data of an ET 200S, the block uses the following inputs of data type BYTE with the meaning shown below:

- SUBN1_ID = Primary ID of the master system
- SUBN2_ID = Secondary ID of the master system
- RACK_NO = Station number (address) of the DP master system of the link
- PADP_ADR = Station number (address) of the ET 200S
- SLAVE _NO = 1st module address of the ET 200S in the link
- SLAVE_SL = Number of slots at the ET 200S

The "Generate module drivers" CFC function fetches this data from HW Config.

The useful diagnostic data contains information about the ET 200S status.

The status of an ET 200S module is entered in byte DPA_M_xx.
An ET 200S can have up to 64 slots (modules).

Bits 0 to 2 of DPA_M are defined as follows:

| Status Bit 2 | Status Bit 1 | Status Bit 0 | Meaning |
|---|---|---|---|
| 0 | 0 | 0 | Module x OK (valid user data) |
| 0 | 1 | 0 | Module x error (invalid user data) |
| 0 | 0 | 1 | Wrong module x (invalid user data) |
| 0 | 1 | 1 | No module x (invalid user data) |
| 1 | x | x | ET 200S failure (invalid user data) |

Note: If you want to change the SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs online, you must set input ACC_ID = TRUE. This verifies the Link states and updates output values.

### Redundancy

Only non-redundant devices may be used downstream of a Y link.

### Error handling

The validity of input parameters is not checked.

### Startup characteristics

The system verifies that the ET 200S is available.

### Overload behavior

The overload behavior takes place in the upstream OB_DIAG1 block.

### Time response

Not available

### Message response

Not available

### Operator control and monitoring

The block has no faceplate.

## 3.6.2 I/Os of DPDIAGV0

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **DADDR** | Diagnostic address of the Y-Link | INT | 0 |
| **PADP_ADR** | DP address ET 200S | BYTE | 255 |
| **RACK_NO** | Rack/station number | BYTE | 0 |
| **SLAVE_NO** | 1. Slot number of the slave in the Y-link | BYTE | 0 |
| **SLAVE_SL** | Number of ET 200S slots | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DPA_M_xx | Status of the DP/PA slave (xx = 00 - 63) | BYTE | 0 |
| EN_Mxx | 1 = Enable modules (xx = 00 - 63) | BOOL | 0 |
| **QRACKF** | 1 = ET 200S failure | BOOL | 0 |
| **SUBNERR** | 1 = Y link failure | BOOL | 0 |

### In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **ACC_ID** | 1 = accept MODE settings | BOOL | 0 |
| **CPU_DIAG** | CPU diagnostics (system structure) | STRUCT | |
| **RAC_DIAG** | Rack diagnostics of the DP slave downstream of Y link (system structure) | STRUCT | |

# 3.7 DREP: Diagnostic Repeater in the DP master system

## 3.7.1 Description of DREP

### Object name (type + number)

FB 113

- DREP block I/Os (Page 55)

### Area of application

The DREP block evaluates the diagnostic data from a SIMATIC diagnostic repeater for PROFIBUS DP. This repeater must be connected to a DP master.

### Calling OBs

| | |
|---|---|
| OB 1 | Cyclic processing |
| OB 82 | Diagnostic interrupt |
| OB 86 | Rack failure |
| OB 100 | Warm restart (startup, message initialization) |

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- Block OB_DIAG1 is installed in the run sequence upstream of the DREP block.
- The following addresses are configured:
  – The diagnostic address DADDR of the diagnostics repeater
  – The geographic address (SUBN_ID and PADP_ADR)
- The following I/Os are interconnected:
  – The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the OB_DIAG1 block with the IN_OUT structures of the same name of the DREP block.
  – The EN input with the output of an AND block.
  – The inputs of the AND block with the EN_SUBx outputs (x = number of the DP master system) of the OB_BEGIN block, with EN_Rxxx (xxx = rack/station number) of the SUBNET block and with EN_F of the OB_DIAG1 block.
  – EN_DIAG with the EN_DIAG output of the OB_DIAG1 block.

## Function and method of operation

The diagnostics repeater is assigned the following tasks:

- Diagnostics function for two PROFIBUS segments (DP2 and DP3):

  The diagnostics function returns the location and cause of line faults, such as a wire break or missing terminating resistors.

  The error location is output, including a reference to the relevant nodes, e.g., "Short-circuit to shielding at signal line A, node 12 <-> 13".
- Repeater function for three PROFIBUS segments (DP1, DP2, DP3):

  The diagnostics repeater amplifies data signals on the bus and interconnects the relevant RS-485 segments.
- Galvanic/electrical isolation of the PG interface from other bus segments:

  Even if the system is operating at higher transmission rates, interference due to the removal or connection of PG cables is not to be expected at the other PROFIBUS DP segments.

The manual titled *Diagnostic Repeaters for PROFIBUS DP* contains information about the structure of line error states at the DP1, DP2 and DP3 segments and describes the PG interface.

Block DREP reports only the diagnostic events at segments DP2 and DP3 of the diagnostic repeater.

Events of the DP1 segment are reported as general "Cable disturbance" group error.

The PG interface is not evaluated and does not result in a message.

Failure and return of the diagnostics repeater are detected by the upstream OB_DIAG1 block, and are forwarded to the block to report "DR failure".

If an error occurs, an incoming "line error" group message is generated for each segment (DP2 or DP3) when a diagnostic repeater detects the error event (bits in the diagnostic message frame indicating the cause of error):

| Bit | Description |
|---|---|
| A.0 | 1: The location and cause of the error cannot be identified (possibly electromagnetic interference) |
| A.1 | CPU redundancy loss |
| A.2 | 1: - - |
| A.3 | 1: Further measurement circuits at the segment, the other diagnostic repeater is connected to its segment DP2 |
| A.4 | 1: Further measurement circuits at the segment, the other diagnostic repeater is connected to its segment DP3 |
| A.5 | 1: - - |
| A.6 | 1: Cause of error is not clearly identified |
| A.7 | 1: Critical message frame error rate |
| B.0 | 1:- -. |
| B.1 | 1: - - |
| B.2 | 1: - - |

| Bit | Description |
| --- | --- |
| B.3 | 1: - - |
| B.4 | 1: - -. |
| B.5 | 1: - - |
| B.6 | 1: - -. |
| B.7 | 1: - - |
| C.0 | 1: Segment automatically switched off due to continuous zero level on the line. |
| C.1 | 1: Segment automatically switched off due to constantly fluctuating line levels. |
| C.2 | 1: - - |
| C.3 | 1: - - |
| C.4 | 1: More than 32 nodes connected to the measurement segment. |
| C.5 | 1: The distance between the node and the diagnostic repeater exceeds the permitted line length. |
| C.6 | 1: The maximum permitted number of diagnostic repeaters connected in series has been exceeded. |
| C.7 | 1: - - |

The outgoing message will be generated when all segment bits are equal to zero.

Call HW Config to analyze details of events output by the diagnostic repeater.

An appropriate incoming/outgoing message will be generated for each segment (DP2 or DP3) in response to the following events detected by a diagnostic repeater:

| Bit | Description |
| --- | --- |
| A.0 | 1: - |
| A.1 | 1:- |
| A.2 | 1: - |
| A.3 | 1: - |
| A.4 | 1: - |
| A.5 | 1: - |
| A.6 | 1: - |
| A.7 | 1: |
| B.0 | 1: Wire break on signal line A. |
| B.1 | 1: Short-circuit to shield on signal line B. |
| B.2 | 1: - |
| B.3 | 1: Short-circuit to shield on signal line A. |
| B.4 | 1: Wire break on signal line B. |
| B.5 | 1: - |
| B.6 | 1: Wire break on signal line A and/or B, or the terminating resistor is missing. |
| B.7 | 1: Short-circuit between signal line A and/or B, or an additional terminating resistor has been installed. |
| C.0 | 1: - |
| C.1 | 1: - |
| C.2 | 1: - |
| C.3 | 1: - |

| Bit | Description |
|---|---|
| C.4 | 1: - |
| C.5 | 1: -. |
| C.6 | 1: - |
| C.7 | 1: - |

Events detected by the diagnostics repeater are acquired synchronously in OB 82. Diagnostic event data is fetched via SFB 54 in the OB_BEGIN block and written to the structure DINFO. The function always sets just one bit to indicate the cause of an event-entering state. Bit C7 may also be set if the diagnostics repeater has detected further errors. In this case, all previously reported events will be queued. DREP generates a corresponding group error message via ALARM_8P. Flutter messages may occur, particularly in response to error causes A.0.1 and A.6.1. They are suppressed as follows:

After an outgoing message, a new outgoing message will be delayed by the time in [s] set at the DELAY parameter. If a further error is queued, the outgoing message will not be generated until this error has been reported outgoing.

## Error handling

The block evaluates the error information from ALARM_8P, and writes it to the corresponding output parameters. You will find additional information in the "Error information of output parameter MSG_STAT (Page 398)" section.

The block reports a diagnostic error if an error occurs while reading the diagnostic data, or if any other fault corrupts diagnostic data.

## Startup characteristics

ALARM_8P messages are initialized by the DREP block. This uses SFC13 (DPNRM_DG) to read the latest diagnostic information from the diagnostic repeater.

## Overload behavior

The interconnected OB_DIAG1 locks the call of DREP for diagnostics if an overload has occurred.

## Dynamic response:

Not available

## Message response:

The multiple instances ALARM_8P are only called if a message is to be output by this instance. It is only at this point that previously acknowledged messages are updated by the corresponding ALARM block. If the connection to WinCC is down, each ALARM_8P instance can hold up to two message statuses of its event ID. (Usually two messages maximum). Flutter messages can be suppressed via the DELAY input.

The block generates the messages listed below:

| OB no. | Start Event | Message |
|---|---|---|
| 1 | Cyclic processing | Call of ALARM_8P due to incomplete transfer or unacknowledged message |
| 82 | Diagnostic interrupt | Group message |
| 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the sections:

Message texts and associated values of DREP (Page 57)

Maintenance status of MS (Page 403)

## 3.7.2 I/Os of DREP

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| **DADDR** | Diagnostic address of the diagnostic repeater | INT | 0 |
| DELAY | Interrupt delay (s) | INT | 2 |
| **EN_DIAG** | 1 = Queued diagnostic event | BOOL | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **PADP_ADR** | DP/PA address of the diagnostic repeater | BYTE | 255 |
| **SUBN_ID** | DP master system ID | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Type | Default | Type |
|---|---|---|---|---|
| DINFO | Diagnostic information from the diagnostic repeater | STRUCT | | O |
| MSG_ACKx | Message acknowledgment | WORD | 0 | O |
| MSG_STATx | Message error information | WORD | 0 | O |
| O_MS | Maintenance status | DWORD | 0 | O |
| **QDREPF** | 1 = Removed/faulty diagnostic repeater | BOOL | 0 | O |
| QERR | 1 = program error | BOOL | 1 | O |

### In-out parameters

| I/O (parameter) | Meaning | Type | Default | Type |
|---|---|---|---|---|
| **ACC_MODE** | 1 = accept MODE settings | BOOL | 0 | IO |
| **CPU_DIAG** | CPU diagnostics | STRUCT | | IO |
| RAC_DIAG | OB_DIAG1 diagnostics | STRUCT | | IO |

## Additional information

For additional information, refer to the sections:

Message texts and associated values of DREP (Page 57)

Maintenance status of MS (Page 403)

## 3.7.3 Message texts and associated values of DREP

### Assignment of message text and message class (Page 401)

| Message block ALARM_8P | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | DR @1%d@/@2%d@/ Segment DP2: Line fault | S |
| | 2 | DR @1%d@/@2%d@/DP2: → @5%d@.@6%d@m/wire break A /@3%d@ ↔ @4%d@ | S |
| | 3 | DR @1%d@/@2%d@/DP2: → @5%d@.@6%d@m/short A/@3%d@ ↔ @4%d@ | S |
| | 4 | DR @1%d@/@2%d@/DP2: → @5%d@.@6%d@m/wire break B/@3%d@↔@4%d@ | S |
| | 5 | DR @1%d@/@2%d@/DP2: → @5%d@.@6%d@m/short B/@3%d@↔@4%d@ | S |
| | 6 | DR @1%d@/@2%d@/DP2: → @5%d@.@6%d@m/wire break AB or terminating resistor missing/@3%d@↔@4%d@ | S |
| | 7 | DR @1%d@/@2%d@/DP2: → @5%d@.@6%d@m/short AB or too many terminating resistors/@3%d@↔@4%d@ | S |
| | 8 | DR @1%d@/@2%d@/ segment DP1: Line fault | S |
| EV_ID2 | 1 | DR @1%d@/@2%d@/ segment DP3: Line fault | S |
| | 2 | DR @1%d@/@2%d@/DP3: → @5%d@.@6%d@m/wire break A /@3%d@ ↔ @4%d@ | S |
| | 3 | DR @1%d@/@2%d@/DP3: → @5%d@.@6%d@m/short A/@3%d@ ↔ @4%d@ | S |
| | 4 | DR @1%d@/@2%d@/DP3: → @5%d@.@6%d@m/wire break B/@3%d@↔@4%d@ | S |
| | 5 | DR @1%d@/@2%d@/DP3: → @5%d@.@6%d@m/short B/@3%d@↔@4%d@ | S |
| | 6 | DR @1%d@/@2%d@/DP3: → @5%d@.@6%d@m/wire break AB or terminating resistor missing/@3%d@↔@4%d@ | S |
| | 7 | DR @1%d@/@2%d@/DP3: → @5%d@.@6%d@m/short AB or too many terminating resistors/@3%d@↔@4%d@ | S |
| | 8 | DR @1%d@/@2%d@: @7W%t#DREP_TXT@ | S |

You will find the message texts and their text numbers in "Text library for DREP (Page 416)".

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID1 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | PADP_ADR | Address of diagnostic repeater (byte) |
| | 3 | | Station x (segment DP2) |
| | 4 | | Station y (segment DP2) |
| | 5 | | Removal of diagnostic repeater (segment DP2) |
| | 6 | | Removal of diagnostic repeater (segment DP2) |
| EV_ID2 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | PADP_ADR | Address of diagnostic repeater (byte) |
| | 3 | | Station x (segment DP3) |
| | 4 | | Station y (segment DP3) |
| | 5 | | Removal of diagnostic repeater (segment DP3) |
| | 6 | | Removal of diagnostic repeater (segment DP3) |
| | 7 | | Text number (Message 1 - 2) of DREP_TXT |

# 3.8 DREP_L: Diagnostic Repeater downstream of a Y-Link

## 3.8.1 Description of DREP_L

### Object name (type + number)

FB 125

- DREP_L Block I/Os (Page 64)

### Area of application

The DREP_L block evaluates diagnostic data from a SIMATIC diagnostic repeater for PROFIBUS DP. The diagnostic repeater (after DPV0) must be connected downstream of a Y-Link (after DPV1).

### Calling OBs

| | |
|---|---|
| OB 1 | Cyclic processing |
| OB 82 | Diagnostic interrupt |
| OB 86 | Rack failure |
| OB 100 | Warm restart (startup, message initialization) |

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The OB_DIAG1 block is integrated into the run sequence upstream of DREP_L.
- The following addresses are configured:
  - The diagnostic address of the DP/PA link (DADDR) is connected downstream of the diagnostic repeater
  - The geographical address (SUBN1_ID, SUBN2_ID, RACK_NO and PADP_ADR)
- The following I/Os are interconnected:
  - The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the OB_DIAG1 block with the DREP_L INOUT structures of the same name.
  - The EN input is interconnected with the output of an AND block.
    The inputs of the AND block with the EN_SUBx outputs (x = number of the DP master system) of the OB_BEGIN block, with EN_Rxxx (xxx = rack/station number) of the SUBNET block, and with EN_F of the OB_DIAG1 block.
  - EN_DIAG with the EN_DIAG output of the OB_DIAG1 block.

## Function and method of operation

The diagnostics repeater is assigned the following tasks:

- Diagnostics function for two PROFIBUS segments (DP2 and DP3):

  The diagnostics function returns the location and cause of line faults, such as a wire break or missing terminating resistors.

  The error location is output, including a reference to the relevant nodes, e.g., "Short-circuit to shielding at signal line A, node 12 <-> 13".
- Repeater function for three PROFIBUS segments (DP1, DP2, DP3):

  The diagnostics repeater amplifies data signals on the bus and interconnects the relevant RS-485 segments.
- Galvanic/electrical isolation of the PG interface from other bus segments:

  Even if the system is operating at higher transmission rates, interference due to the removal or connection of PG cables is not to be expected at the other PROFIBUS DP segments.

The manual titled *Diagnostic Repeaters for PROFIBUS DP* contains information about the structure of line error states at the DP1, DP2 and DP3 segments and describes the PG interface.

Block DREP_L only reports diagnostic events at segments DP2 and DP3 of the diagnostic repeater.

Events of the DP1 segment are reported as general "Cable disturbance" group error.

The PG interface is not evaluated and does not result in a message.

Failure and return of the diagnostics repeater are detected by the upstream OB_DIAG1 block, and are forwarded to the block to report "DR failure".

If an error occurs, an incoming "line error" group message is generated for each segment (DP2 or DP3) when a diagnostic repeater detects the error event (bits in the diagnostic message frame indicating the cause of error):

| Bit | Description |
|---|---|
| A.0 | 1: The location and cause of the error cannot be identified (possibly electromagnetic interference) |
| A.1 | CPU redundancy loss |
| A.2 | 1: - - |
| A.3 | 1: Further measurement circuits at the segment, the other diagnostic repeater is connected to its segment DP2 |
| A.4 | 1: Further measurement circuits at the segment, the other diagnostic repeater is connected to its segment DP3 |
| A.5 | 1: - - |
| A.6 | 1: Cause of error is not clearly identified |
| A.7 | 1: Critical message frame error rate |
| B.0 | 1: |
| B.1 | 1: |
| B.2 | 1: - - |

| Bit | Description |
|---|---|
| B.3 | 1: |
| B.4 | 1: |
| B.5 | 1: - - |
| B.6 | 1: |
| B.7 | 1: |
| C.0 | 1: Segment automatically switched off due to continuous zero level on the line. |
| C.1 | 1: Segment automatically switched off due to constantly fluctuating line levels. |
| C.2 | 1: - - |
| C.3 | 1: - - |
| C.4 | 1: More than 32 nodes connected to the measurement segment |
| C.5 | 1: The distance between the node and the diagnostic repeater exceeds the permitted line length |
| C.6 | 1: The maximum permitted number of diagnostic repeaters connected in series has been exceeded |
| C.7 | 1: - - |

The outgoing message will be generated when all segment bits are equal to zero.

Call HW Config to analyze details of events output by the diagnostic repeater.

An appropriate incoming/outgoing message will be generated for each segment (DP2 or DP3) in response to the following events detected by a diagnostic repeater:

| Bit | Description |
|---|---|
| A.0 | 1: - |
| A.1 | 1:- |
| A.2 | 1: - |
| A.3 | 1: - |
| A.4 | 1: - |
| A.5 | 1: - |
| A.6 | 1: - |
| A.7 | 1: |
| B.0 | 1: Wire break at signal line A |
| B.1 | 1: Short-circuit to shielding at signal line B |
| B.2 | 1: - |
| B.3 | 1: Short-circuit to shielding at signal line A |
| B.4 | 1: Wire break at signal line B |
| B.5 | 1: - |
| B.6 | 1: Wire break at signal line A and/or B, or the terminating resistor is missing |
| B.7 | 1: Short-circuit signal line A <-> B, or an additional terminating resistor has been installed |
| C.0 | 1: - |
| C.1 | 1: - |
| C.2 | 1: - |
| C.3 | 1: - |

| Bit | Description |
|---|---|
| C.4 | 1: - |
| C.5 | 1: -. |
| C.6 | 1: - |
| C.7 | 1: - |

Events detected by the diagnostics repeater are acquired synchronously in OB 82. Diagnostic event data is fetched via SFB 54 in the OB_BEGIN block and written to the structure DINFO. The function always sets only one bit to indicate the cause of an incoming event. Bit C7 may also be set if the diagnostics repeater has detected further errors. In this case, all previously reported events will be queued. DREP_L generates a corresponding group error message via ALARM_8P. Flutter messages may occur, particularly in response to error causes A.0.1 and A.6.1. They are suppressed as follows:

After an outgoing message, a new outgoing message will be delayed by the time in [s] set at the DELAY parameter. If a further fault is queued, the outgoing message will not be generated until this fault is outgoing.

## Error handling

The block evaluates the error information from ALARM_8P, and writes it to the corresponding output parameters.

You will find additional information in the "Error information of output parameter MSG_STATx (Page 398)" section.

The block reports a diagnostic error if an error occurs while reading the diagnostic data, or if any other fault corrupts diagnostic data.

## Startup characteristics

ALARM_8P messages are initialized by the DREP_L block The current diagnostics information is read from the diagnostics repeater using SFB 52 (RDREC).

## Overload behavior

In the event of an overload, the upstream OB_DIAG1 block prevents DREP_L being called for diagnostics.

## Dynamic response:

Not available

## Message response:

The multiple instances ALARM_8P are only called if a message is to be output by this instance. It is only at this point that previously acknowledged messages are updated by the corresponding ALARM block. If the connection to WinCC is down, each ALARM_8P instance can hold up to two message statuses of its event ID. Flutter messages can be suppressed via the DELAY input.

The block generates the messages listed below:

| OB no. | Start Event | Message |
|---|---|---|
| 1 | Cyclic processing | Call of ALARM_8P due to incomplete transfer or unacknowledged message |
| 82 | Diagnostic interrupt | Group message |
| 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message Texts and associated values of DREP_L (Page 66)

Maintenance status of MS (Page 403)

## 3.8.2 I/Os of DREP_L

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Type | De-fault |
|---|---|---|---|
| **DADDR** | Diagnostic address of the DP/PA link | INT | 0 |
| DELAY | Interrupt delay (s) | INT | 2 |
| **EN_DIAG** | 1 = Queued diagnostic event | BOOL | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **PADP_ADR** | DP/PA address of the diagnostic repeater | BYTE | 255 |
| **RACK_NO** | Number of the rack | BYTE | 255 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Type | De-fault | Type |
|---|---|---|---|---|
| DINFO | Diagnostic information from the diagnostic repeater | STRUCT | | O |
| MSG_ACKx | Message acknowledgment | WORD | 0 | O |
| MSG_STATx | Message error information | WORD | 0 | O |
| O_MS | Maintenance status | DWORD | 0 | O |
| **QDREPF** | 1 = Removed/faulty diagnostic repeater | BOOL | 0 | O |
| QERR | 1 = program error | BOOL | 1 | O |

## In-out parameters

| I/O (parameter) | Meaning | Type | De-fault | Type |
|---|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 | IO |
| CPU_DIAG | CPU diagnostics | STRUCT | | IO |
| RAC_DIAG | OB_DIAG1 diagnostics | STRUCT | | IO |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of DREP_L (Page 66)

Maintenance status of MS (Page 403)

### 3.8.3 Message texts and associated values of DREP_L

#### Assignment of message text and message class

| Message block ALARM_8P | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | DR @1%d@/@2%d@/@3%d@/ Segment DP2: Line fault | S |
| | 2 | DR @1%d@/@2%d@/@3%d@/DP2: → @6%d@.@7%d@m/wire break A/ @4%d@ ↔ @5%d@ | S |
| | 3 | DR @1%d@/@2%d@/@3%d@/DP2: → @6%d@.@7%d@m/short A/@4%d@ ↔ @5%d@ | S |
| | 4 | DR @1%d@/@2%d@/@3%d@/DP2: → @6%d@.@7%d@m/wire break B/@4%d@ ↔ @5%d@ | S |
| | 5 | DR @1%d@/@2%d@/@3%d@/DP2: → @6%d@.@7%d@m/short B/@4%d@ ↔ @5%d@ | S |
| | 6 | DR @1%d@/@2%d@/@3%d@/DP2: → @6%d@.@7%d@m/wire break AB or terminating resistor missing/@4%d@ ↔ @5%d@ | S |
| | 7 | DR @1%d@/@2%d@/@3%d@/DP2: → @6%d@.@7%d@m/short AB or too many terminating resistors/@4%d@ ↔ @5%d@ | S |
| | 8 | DR @1%d@/@2%d@/@3%d@/ Segment DP1: Line fault | S |
| EV_ID2 | 1 | DR @1%d@/@2%d@/@3%d@/ Segment DP3: Line fault | S |
| | 2 | DR @1%d@/@2%d@/@3%d@/DP3: → @6%d@.@7%d@m/wire break A/ @4%d@ ↔ @5%d@ | S |
| | 3 | DR @1%d@/@2%d@/@3%d@/DP3: → @6%d@.@7%d@m/short A/@4%d@ ↔ @5%d@ | S |
| | 4 | DR @1%d@/@2%d@/@3%d@/DP3: → @6%d@.@7%d@m/wire break B/@4%d@ ↔ @5%d@ | S |
| | 5 | DR @1%d@/@2%d@/@3%d@/DP3: → @6%d@.@7%d@m/short B/@4%d@ ↔ @5%d@ | S |
| | 6 | DR @1%d@/@2%d@/@3%d@/DP3: → @6%d@.@7%d@m/wire break AB or terminating resistor missing/@4%d@ ↔ @5%d@ | S |
| | 7 | DR @1%d@/@2%d@/@3%d@/DP3: → @6%d@.@7%d@m/short AB or too many terminating resistors/@4%d@ ↔ @5%d@ | S |
| | 8 | DR @1%d@/@2%d@/@3%d@: @8W%t#DREP_L_TXT@ | S |

You will find the message texts and their text numbers in "Text library for DREP_L (Page 416)".

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID1 | 1 | SUBN_ID1 | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | Address of diagnostic repeater (byte) |
| | 4 | | Station x (segment DP2) |
| | 5 | | Station y (segment DP2) |
| | 6 | | Removal of diagnostic repeater (segment DP2) |
| | 7 | | Removal of diagnostic repeater (segment DP2) |
| EV_ID2 | 1 | SUBN_ID1 | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | Address of diagnostic repeater (byte) |
| | 4 | | Station x (segment DP3) |
| | 5 | | Station y (segment DP3) |
| | 6 | | Removal of diagnostic repeater (segment DP3) |
| | 7 | | Removal of diagnostic repeater (segment DP3) |

If SUBN_ID1 = 16#FF, SUBN_ID2 is used as associated value 1.

## See also

Message Classes (Page 401)

# 3.9 FFD_CIF: Monitoring FF slaves in CIF mode

## 3.9.1 Description of FFD_CIF

### Object name (type + number)

FB 145

- I/Os of FFD_CIF (Page 70)

### Area of application and method of operation

The FFD_CIF blocks monitors the higher-level errors of a field device, such as failure/return. The block runs acyclically.

The block FFD_CIF is enabled to run by the higher-level DPAY_V1 block. Start and diagnostic information is read from the CPU_DIAG structure. The event to be evaluated is entered in the start information (CPU_DIAG) of OB_BEGIN. The block FFD_CIF checks the geographic address and SLOT_NO of the FF device to determine whether it is responsible for this event.

Byte 3 of the additional alarm information contains the slot number of the field device that triggered the diagnostic interrupt. The corresponding slot is enabled.

The parameter EN_MSG_D = 1 is still set here. This means that the AS asset faceplate for the FF device accesses this block; messages and maintenance state are generated here.

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The FFD_CIF block is installed downstream from the DPAY_V1 block.
- The RACK_NO, DADDR, EN_MSG_D, SUBN1_ID, SUBN2_ID, SLOT_NO and FFDP_ADR inputs are configured.
- The EN input is interconnected with the output of an AND block. These block inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block, with EN_Mx (x= number of the FF device) of the DPAY_V1 block and EN_F of the OB_DIAGF block.
- The CPU_DIAG OUT structure of the OB_BEGIN block and SUB_DIAG of the SUBNET block are interconnected with the IN_OUT structures of the FFD_CIF block.
- RAC_DIAG is interconnected with the OUT structure RAC_DIAG of OB_DIAGF block.
- DINFO is interconnected with the OUT structure DINFO of the OB_DIAGF block.
- The OUT structure FF_DIAG of OB_DIAGF is interconnected with the IN_OUT structure of the same name of FFD_CIF.

## Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 72 | CPU redundancy loss |
| OB 70 | Redundancy loss |
| OB 82 | Diagnostic interrupt |
| OB 83 | Remove/insert module interrupt (failure/return of a field device) |
| OB 86 | Rack failure |
| OB 100 | Restart |

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Message response

The DELAY input is used to delay the outputting of error messages for an outgoing, higher-priority error. For example, if the FFD_CIF block recognizes an outgoing error at a DP master connected to it, it initially assumes that there is a faulty assigned DP slave in the rack it monitors, and sets the corresponding output SUBNxERR. The error status is not reset until the DP slave returns (in this case: OB 86, OB 70). The FFD_CIF blocks suppress the potential slave failure states for DELAY seconds so as not to trigger a surge of messages from DP slaves which are not yet synchronized when the master returns. An error message is only output to the OS when the DP slave has not reported its return before this delay time has expired.

**Note:** Do not set the value of DELAY too high, otherwise DP slaves that were removed during the master failure or are defective will be signaled to the OS too late after the DP master returns.

## 3.9.2 I/Os of FFD_CIF

### I/Os of FFD_CIF

The factory setting of the block display in CFC is identified in the "I/O" column.

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **DADDR** | Diagnostic address of the FF link | INT | 0 |
| DELAY | Interrupt delay (s) | INT | 15 |
| **DINFO** | Diagnostic information | STRUCT | |
| EN_MSG | 1 = enable interrupt | BOOL | 1 |
| EN_MSG_D | 1 = Enable message "Device failure" | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **FFDP_ADR** | FF slave address | BYTE | 16#FF |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 16#FF |
| **SLOT_NO** | Slot number | BYTE | 16#FF |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 16#FF |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 16#FF |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Error message status | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE | Channel 0 mode | DWORD | 0 |
| **QMODF** | 1 = module removed/defective | BOOL | 0 |
| **QRACKF** | 1 = Slave failure/faulty | BOOL | 0 |
| **RETURN_CODE** | Return value of the function | INT | |
| **SUBN1ACT** | 1 = Slave 1 is active | BOOL | 0 |
| **SUBN2ACT** | 1 = Slave 2 is active | BOOL | 0 |
| **SUBN1ERR** | 1 = Error in the primary DP master system | BOOL | 0 |
| **SUBN2ERR** | 1 = Error in the redundant DP master system | BOOL | 0 |
| V1_MODE | 1 = DPV1 mode of the DP master system | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| FF_DIAG | Diagnostics of the FF device | STRUCT | |
| RAC_DIAG | System structure RACK diagnostics | STRUCT | |
| SUB_DIAG | System structure: SUBNET diagnostics | STRUCT | |

## 3.9.3 Message texts and associated values of FFD_CIF

### Message texts and associated values of FFD_CIF

Assignment of message text and message class (Page 401)

| Message block | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 (ALARM_8P) | 1 | Device @1%d@/ @2%d@/@3%d@: bad, maintenance alarm | S |
| | 2 | Device @1%d@/ @2%d@/@3%d@: uncertain, maintenance request | F |
| | 3 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 83) | S |
| | 4 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 82) | S |
| | 5 | Device @1%d@/ @2%d@/@3%d@: good, maintenance required | M |
| | 6 | | No message |
| | 7 | Device @1%d@/ @2%d@/@3%d@: Configuration error: Name does not match address | S |
| | 8 | Device @1%d@/ @2%d@/@3%d@: Failure | S |

### Assignment of Associated Values

| Associated value | Block parameters |
|---|---|
| 1 | DP master system ID (SUBN_ID) |
| 2 | Rack/station number (RACK_NO) |
| 3 | FF slave address (FFDP_ADR) |

# 3.10 FFDP_L1: Monitoring FF slaves with maximum 32 values

## 3.10.1 Description of FFDP_L1

### Object name (type + number)

FB 139

- I/Os of FFDP_L1 (Page 75)

### Area of application and method of operation

The FFDP_L1 block monitors the higher-level errors of a field device, such as failure/return and runs acyclically.

The FFDP_L1 block is enabled to run by the higher-level DPAY_V1 block. The event to be evaluated is entered in the start information (CPU_DIAG) of OB_BEGIN. The FFDP_L1 block checks the geographic address and SLOT_NO of the FF device to determine whether it is responsible for this event.

Byte 3 of the additional alarm information contains the slot number of the field device that triggered the diagnostic interrupt. The corresponding slot is enabled.

If a higher-level error goes away the diagnostics is read per SFC 13 from FF link (the data are all located locally on the link and do not have to be retrieved by the FF device) and reported accordingly.

The FFDP_L1 block monitors

- FF devices downstream from an FF link (DPV1 slave)

It generates the MODE (FF_MODE) and the value status for the signal processing blocks. The subordinate blocks FF_MOD32 are always enabled for processing and run cyclically. The information is relayed via the structure FF_DIAG to the FF_MOD32. This structure outputs the message.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 72 | CPU redundancy loss |
| OB 70 | Redundancy loss |
| OB 82 | Diagnostic interrupt |
| OB 83 | Remove/insert module interrupt (failure/return of a field device) |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Restart |

## Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The FFDP_L1 block is installed downstream from the DPAY_V1 block.
- The CPU_DIAG OUT structure of the OB_BEGIN block and SUB_DIAG of the SUBNET block are interconnected with the IN_OUT structures of the same name of block FFDP_L1.
- RAC_DIAG is interconnected with the OUT structure RAC_DIAG of OB_DIAGF block.
- The following will be configured:
  - Diagnostic address of the FF link DADDR
  - Geographical address (SUBN1_ID, SUBN2_ID, RACK_NO) of the FF device
  - SLOT_NO
  - FF address of the FF device (FFDP_ADR)
  - MODE_xy (mode of the FF device)
  - NUM_IODATA
- The EN input is interconnected with the output of an AND block. These block inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block, with EN_Mx (x= number of the FF device) of the DPAY_V1 block and EN_F of the OB_DIAGF block.
- The QPERAF output is interconnected with the PERAF input of the FF_MOD32.
- The QMODF output is interconnected with the MODF input of the FF_MOD32.
- The FF_DIAG output is interconnected with the FF_DIAG input of the FF_MOD32.
- DINFO is interconnected with the OUT structure DINFO of the OB_DIAGF block.

## Message response

The DELAY input is used to delay the outputting of error messages for an outgoing, higher-priority error. For example, if the FFDP_L1 block recognizes an outgoing error at a DP master connected with it, it initially assumes that there is a faulty assigned DP slave in the rack it monitors, and sets the corresponding output SUBNxERR. The error status is not reset until the DP slave returns (in this case: OB 86, OB 70). The FFDP_L1 blocks suppress the potential slave failure states for DELAY seconds so as not to trigger a surge of messages from DP slaves which are not yet synchronized when the master returns. An error message is only output to the OS when the DP slave has not reported its return before this delay time has expired.

Note: Do not set the value of DELAY too high, otherwise DP slaves that were removed during the master failure or are defective will be signaled to the OS too late after the DP master returns.

## 3.10.2 I/Os of FFDP_L1

### I/Os of FFDP_L1

The factory setting of the block display in CFC is identified in the "I/O" column.

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **DADDR** | Diagnostic address of the FF link | INT | 0 |
| DELAY | Interrupt delay (s) | INT | 15 |
| **FFDP_ADR** | Address of the FF device | BYTE | 0 |
| MODE | Device status | WORD | 0 |
| **NUM_IODATA** | Number of I/O data in the FF device | INT | 0 |
| **RACK_NO** | Number of the rack | BYTE | 0 |
| **SLOT_NO** | Number of the slot | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 16#FF |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 16#FF |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **EN_D_Q** | 1 = Processing of the quality code | BOOL | 0 |
| FF_DIAG | FF device diagnostics | DWORD | 0 |
| OMODE | Operating mode slot | DWORD | 0 |
| QERR | 1 = Error runtime | BOOL | 1 |
| **QMODF** | 1 = Error/defect field device | BOOL | 0 |
| **QPERAF** | 1 = Access error I/O | BOOL | 0 |
| **QRACKF** | 1 = Slave failure/faulty | BOOL | 0 |
| **RAC_DIAG** | System structure RACK diagnostics | STRUCT | |
| RETURN_CODE_O | Return value | INT | 0 |
| **SUBN1ACT** | 1 = Slave 1 is active | BOOL | 0 |
| **SUBN2ACT** | 1 = Slave 2 is active | BOOL | 0 |
| **SUBN1ERR** | 1 = Error in the primary DP master system | BOOL | 0 |
| **SUBN2ERR** | 1 = Error in the redundant DP master system | BOOL | 0 |
| V1_MODE | DPV1 mode of the DP master system | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | CPU diagnostics | STRUCT | |
| DINFO | Diagnostic information | STRUCT | |
| FF_DIAG_I | Diagnostics of the FF device | STRUCT | |
| SUB_DIAG | System structure: SUBNET diagnostics | STRUCT | |

# 3.11 FF_MOD32: Diagnostics of an FF slave (downstream from FF link DPV1)

## 3.11.1 Description of FF_MOD32

### Object name (type + number)

FB 124

- I/Os of FF_MOD32 (Page 79)

### Area of application and method of operation

The block reports the maintenance status of an FF field device downstream from an FF link. For diagnostics in the Maintenance Station, all error sources will be linked with OR and the worst will be assigned to the MS output. The channel blocks (FbAnIn, FbAnOu, FbDiIn, FbDiOu) will be controlled by quality codes.

The block analyses cyclically all events that affect an FF device and reports the following events:

- Evaluation of the RAC_DIAG info (from FFDP_L1): Device failure
- Evaluation of the FF_DIAG info (from FFDP_L1): Diagnostics

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The block is integrated in the run sequence upstream of the FbAxXx, FbDxXx block.
- The SUBN1_ID, SUBN2_ID, RACK_NO, SLOT_NO, FFDP_ADR and NUM_IODATA inputs are configured.
- The block inputs are interconnected with the following outputs:
  - FF_DIAG with FF_DIAG output of the FFDP_L1 block
  - MODE_xy with OMODE_xy outputs of the FFDP_L1 block
  - MODF, PERAF with QMODF and QPERAF outputs of the FFDP_L1 block
  - RAC_DIAG with output structure RAC_DIAG of the OB_DIAGF block
  - QC_x input with icon of the FF field device
  - EN_D_Q with EN_DQ output of the FFDP_L1 block
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

- The output parameter of OMODE_xy is interconnected with the following channel block at the OMODE_xy parameter.
- The OUT structure CPU_DIAG of the OB_BEGIN block is interconnected with the IN_OUT structures of the same name of the FFDP_L1 block.

## Calling OBs

| OB no. | Start Event | Message |
|---|---|---|
| OB 1 | Cyclic program | |
| OB 100 | Warm restart | The Alarm_8P is initialized at startup (OB 100) |

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## 3.11.2 I/Os of FF_MOD32

### I/Os of FF_MOD32

The factory setting of the block display in CFC is identified in the "I/O" column.

I/O name **bold** = I/O visible; I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_D_Q | 1 = Processing of the quality code | BOOL | 0 |
| EN_MSG | 1 = enable interrupt | BOOL | 1 |
| EV_ID1 | Message number 1 | DWORD | 0 |
| EV_ID2 | Message number 2 | DWORD | 0 |
| **FFDP_ADR** | Address of the FF device | BYTE | 0 |
| FF_DIAG | Diagnostics of the FF device | DWORD | |
| MODE_xx | Device mode (xx = IOData 0 - 31) | DWORD | 0 |
| **MODF** | 1 = field device error/fault | BOOL | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **NUM_IODATA** | Number of I/O data in the FF device | INT | 0 |
| **PERAF** | 1 = I/O access error | BOOL | 0 |
| QC_xx | Status of FF field device (x = channel 0 to 31) | BYTE | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 16#FF |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 16#FF |

## Output parameter

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| B_QC | Quality code, bit-granular | STRUCT | |
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| DXCHG_xx | Bidirectional data exchange channel | DWORD | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| IODATA_EXIST | I/O data available | DWORD | 0 |
| IODATA_OK | I/O data OK | DWORD | 0 |
| MSGSTAT1 | Message error - information 1 | WORD | 0 |
| MSGSTAT2 | Message error - information 2 | WORD | 0 |
| MSG_ACK1 | Message acknowledgment 1 | WORD | 0 |
| OMODE_xx | Value status FF field device (xx = IOData 0 to 31) | DWORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = field device error/fault | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = Slave failure/faulty | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CPU_DIAG | CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: SUBNET diagnostics | STRUCT | |

## 3.11.3 Message texts and associated values of FF_MOD32

### Message texts and associated values of FF_MOD32

Assignment of message text and message class (Page 401)

| Message block | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 (ALARM_8P) | 1 | Device @1%d@/ @2%d@/@3%d@: Bad, maintenance alarm | S |
| | 2 | Device @1%d@/ @2%d@/@3%d@: Uncertain, maintenance request | F |
| | 3 | Device @1%d@/ @2%d@/@3%d@: Good, maintenance required | M |
| | 4 | Device @1%d@/ @2%d@/@3%d@: Access error | S |
| | 5 | | No message |
| | 6 | Device @1%d@/ @2%d@/@3%d@: Failure | S |
| | 7 | Device @1%d@/ @2%d@/@3%d@: Configuration error: Name does not match address | S |
| | 8 | | No message |
| EV_ID2 (NOTIFY_8P) | 1 | Device @1%d@/ @2%d@/@3%d@: Good, assuming fail-safe position | SA |
| | 2 | Device @1%d@/ @2%d@/@3%d@: Good, configuration has been changed | SA |
| | 3 | Device @1%d@/ @2%d@/@3%d@: Uncertain, simulation | SA |
| | 4 | Device @1%d@/ @2%d@/@3%d@: Uncertain, process-related, no maintenance | SA |
| | 5 | Device @1%d@/ @2%d@/@3%d@: Bad, process-related, no maintenance | SA |
| | 6 | Device @1%d@/ @2%d@/@3%d@: Bad, local operation/functional check | SA |
| | 7 | Device @1%d@/ @2%d@/@3%d@: Bad, device passivated | SA |
| | 8 | | No message |

### Assignment of Associated Values

| Associated value | Block parameters |
|---|---|
| 1 | DP master system ID (SUBN_ID) |
| 2 | Rack/station number (RACK_NO) |
| 3 | FF slave address (FFDP_ADR) |

# 3.12 FM_CNT: Programming and controlling FM 350 modules

## 3.12.1 Description of FM_CNT

### Object name (type + number)

FB 126

- FM_CNT Block I/Os (Page 86)

### Area of application

Block FM_CNT is used to configure and control FM 350-1 and FM 350-2 modules. It writes the counter levels, limits and comparison values of the FM 350-2 module.

### Calling OBs

OB 100 and the cyclic OB (100 ms recommended) used for transmitting data.

Also note the assignments (Page 402) to the FM_CO block.

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The block is installed in the cyclic OB upstream of the CH_CNT blocks.
- The block runtime group is installed in OB 100 downstream of MOD_D1.
- The LADDR, FM_MODE, RACK_NO, SUBN1_ID, SUBN2_ID, and SLOT_NO inputs are configured.
- The MODEx inputs are interconnected with the OMODEx outputs of the MOD_D1 block.
- The FM_DATA output structure is interconnected with the structure of the same name of the CH_CNT block.
- The OMODEx output is interconnected with the MODE input of the CH_CNT block.
- The OUT structure CPU_DIAG of OB_BEGIN is interconnected with the IN_OUT structure of the same name of FM_CNT.
- The EN_CO input structure is interconnected with the EN_CO_x output structure of the FM_CO block (x = number of the rack).
- The output ENCO is connected to the input ENCOx_yy of the FM_CO block (x = number of the rack, yy = coordination number).

## Addressing

The logical base address of the module is entered in the LADDR I/O by the CFC driver generator.

## Redundancy

Higher-level block MOD_D1 evaluates the redundancy of DP master systems operating in an H system. Redundancy for two FM 350-1 or FM 350-2 modules is not supported, and must be controlled by the user outside the block.

## MODE Setting

Signal states of the MODE_xx input, or QMODE_xx output of the FM_CNT block are described under the MODE settings.

MODE_xx input parameters are available for up to 8 signal channels. Their default setting is "0" (no signal). For each signal channel xx, the operating mode of the FM 350 module must be set at the MODE_xx input (the CFC driver generator does this for you).

The module recognizes the following modes:

| Term | Coding MODE | Description |
|---|---|---|
| Channel not used | 16#0000 | Channel of the FM 350 not used. |
| Continuous counting | 16#xx01 | The FM 350 counts continuously, starting with the current counter level when the internal gate opens. |
| One-time counting | 16#xx02 | The FM 350 counts from the start value to the end value when the internal gate opens. |
| Periodic counting | 16#xx03 | The FM 350 counts between the start value and the end value when the internal gate opens. |
| Frequency measurement | 16#xx04 | The FM 350 determines the frequency pulse sequence at the input. |
| Speed measurement | 16#xx05 | The FM 350 determines the speed of the device connected to the input. |
| Period duration measurement | 16#xx06 | The FM 350 determines the duration of the pulse sequence at the input. |
| Dosing | 16#xx07 | Four channels of the FM 350-2 are used for dosing. |

The count and measured values can be recorded for the FM 350-2 module, either via the process image(fast update) or via "Read data record" (slower update).
If the count and measured values of a channel in the process image are made available, they have to be in alignment in the process image. The following variants are possible.

| Term | Coding MODE | Description |
|---|---|---|
| Count and measured value are not in the process image | 16#Cxxx | Read count and measured values via data record |
| Only the count value is in the process image | 16#8xxx | Read measured value via data record and count value in the process image |
| Only the measured value is in the process image | 16#4xxx | Read count values via data record and measured value in the process image |
| Count and measured value are in the process image | 16#0xxx | Read count and measured value in the process image |
| Data type DWORD count and measured value | 16#x0xx | Count value before measured value, both of the data type DWORD |
| Data type WORD count value | 16#x1xx | Count value of the the data type WORD before measured value of the data type DWORD |
| Data type WORD measured value | 16#x2xx | Count value of the data type DWORD before measured value of the data type WORD |
| Data type WORD count and measured value | 16#x3xx | Measured value before count value, both of the data type WORD |
| Data type DWORD count and measured value | 16#x8xx | Measured value before count value, both of the data type DWORD |
| Data type WORD count value | 16#x9xx | Measured value of the the data type DWORD before count value of the data type WORD |
| Data type WORD measured value | 16#xAxx | Measured value of the the data type WORD before count value of the data type DWORD |
| Data type WORD count and measured value | 16#xBxx | Count value before measured value, both of the data type WORD |

MODE is formed by the linking the operating mode code, and the access type value.
Example: Count and measured value in "Dosing" mode of the the data type DWORD is not in the process image MODE = 16#C007.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

Whenever the system or FM 350-1/FM 350-2 starts up, the block coordinates the restart with the module. The CMP_VALx parameters are then loaded into the FM 350.

ALARM_8P is initialized.

## Overload behavior

Not available

## Time response

Not available

## Message response

The block reports operating and data errors for the FM 350-1 module, and data errors for FM 350-2 using ALARM_8P. The message function can be disabled by setting EN_MSG = FALSE. The MOD_D1 block reports diagnostic interrupts from the FM 350-1 or FM 350-2.

## Operator control and monitoring

The block has no faceplate.

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

## Additional information

You will find more information in:

Message texts and associated values of FM_CNT (Page 88)

## 3.12.2 I/Os of FM_CNT

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CO_NO | Coordination number for data record reading | INT | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **FM_MODE** | 0 = FM 350-1, >0 = FM 350-2 | BYTE | 0 |
| **LADDR** | Logical address FM 350 | INT | 0 |
| **MODEx** | Mode channel (x = 0 to 7) | DWORD | 0 |
| RACK_NO | Rack number | BYTE | 0 |
| SLOT_NO | Slot number | BYTE | 0 |
| SUBN1_ID | ID of the primary DP master system | BYTE | 255 |
| SUBN2_ID | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **FM_DATA** | Structure FM 350 data | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| **OMODEx** | Mode channel (x = 0 to 7) | DWORD | 0 |
| **QDATA_ERR** | 1 = Data error | BOOL | 0 |
| QDONE | 1 = Write new data | BOOL | 0 |
| QDONE_RD | 1 = New data read | BOOL | 0 |
| **QMODF** | 1 = Error FM 350 | BOOL | 0 |
| **QPARF** | 1 = Module not programmed | BOOL | 0 |
| QRD_ERR | 1 = Error when reading data | BOOL | 0 |
| QWR_ERR | 1 = Error read data | BOOL | 0 |
| STATUS_RD | Read status of data record | DWORD | 0 |
| STATUS_WR | Write status of data record | DWORD | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| EN_CO | Current coordination number | STRUCT | |
| ENCO | Coordination number | BYTE | 0 |
| CPU_DIAG | CPU diagnostics (system structure) | STRUCT | |

## Additional information

Additional information is available in the section:

Message texts and associated values of FM_CNT (Page 88)

### 3.12.3 Message texts and associated values of FM_CNT

#### Assignment of message text and message class

| Message block ALARM_8P | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID | 1 | FM 350 @1%d@/@2%d@/@3%d @data error number@4%d@ | S |
| | 2 | FM 350 @1%d@/@2%d@/@3%d @data error number @5%d@ | S |
| | 3 | | No message |
| | 4 | | No message |
| | 5 | | No message |
| | 6 | | No message |
| | 7 | | No message |
| | 8 | | No message |

#### Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |
| | 4 | DA_ERR_W | Data error number |
| | 5 | OT_ERR_B | Operator error number |

#### See also

Message Classes (Page 401)

## 3.13 IMDRV_TS: Transferring time-stamped process-signal changes

### 3.13.1 Description of IMDRV_TS

#### Object name (type + number)

FB 129

- IMDRV_TS block I/Os (Page 94)

#### Area of application

The IMDRV_TS block transfers time-stamped process signal changes to the MSG_TS blocks, and messages from the interface module (IM) to the OS.

#### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 40 | Hardware interrupt |
| OB 100 | Warm restart |

#### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The IMDRV_TS block is copied from the library and instantiated in a system chart. The block is installed in its runtime group after the RACK block runtime group in the above-mentioned OBs.
- OR_32_TS is always interconnected between MSG_TS and IMDRV_TS.
- The logical address LADDR is configured with the logical address of the IM (diagnostics address). If you operate the DP master system in DPV1 mode, the input address of the IM is entered.

- The RAC_DIAG structure of the RACK block is interconnected with the structure of the same name of the IMDRV_TS block.
- The S_CH_xxx and TINF_xxx inputs of the TRIG_INF structure are set.

  Every signal that is time-stamped by the IM has a unique assignment identified by the slot of the module combined with the corresponding channel number. There are 128 inputs of the WORD data type for 128 signals. The slot number of the relevant digital input module is entered in the more significant byte and the channel number (signal of the digital input module) is entered into the less significant byte. The slot and channel number of the process signals are entered in the block inputs S_CH_xxx.

  Example:

  In HW Config, you have activated time-stamping for the digital signal of channel 10 of a digital input module that is inserted in slot 5 of an ET 200M. The number 16#050A is entered at the first available input S_CH_xxx of the IM_DRV_TS.

  The information about the edge evaluation for the event entering state is stored in the TINF_xxx parameters of the TRIG_INF structure.

  0 means: 0 -> 1 is event entering state
  1 means: 1 -> 0 is event entering state.

## Description of the functions

The IMDRV_TS block reads the messages from the message buffers (up to 15 data records, each with 20 messages) of an IM, and transfers them to a message block for time stamps (MSG_TS). The MSG_TS sends the messages via an ALARM_8P block whose time stamps for its 8 messages are entered in the 1st associated value in an ARRAY of BYTE.

## How it works

- Hardware interrupt (OB 40): The IM generates a hardware interrupt if there are new messages. The time-stamp status, including the number of the IM data record to be fetched and the number of messages in the data record, are fetched from the start information of the process interrupt OB and stored for cyclic processing. The interrupt stack can hold up to 17 process interrupts. If this maximum is exceeded, all new information will be lost. The loss of information is indicated by the "Loss of message at IM (buffer overflow)" message.
- Cyclic processing: If any messages are queued in the stack, SFB 52 (RDREC, read data record) reads the relevant data record (message buffer). If there are several data records to be fetched, it will fetch the record that contains the oldest messages (oldest hardware interrupt). The block instance temporarily stores a maximum of 20 messages of a data record.

  The IM can enter new messages in a data record once it has been read. If all data records are in use, the IM enters "Loss of message at IM (buffer overflow)" (incoming) as the last message in the message buffer. "Loss of message at IM (buffer overflow)" (outgoing) is then entered as the first message in the first free data record. Messages received within the interval between a buffer overflow and the enabling of a record will be lost.
  The slot number/channel number of the stored messages are compared with the input parameters of the block for slot number/channel number (S_CH_xxx). If they match, the message is written to the corresponding output (TS_xxx).
  Errors occurring during data exchange between the block and IM are reported by the ALARM_8P block (for example, an I/O access error).

## Quality code for the time stamp TS_xxx.TS0/TS1

A quality code QC_TS is formed for the time stamp TS0/TS1, and entered in TS_xxx.

| State | Quality code TS_xxx.QC_TS |
|---|---|
| Valid value | 16#80 |
| Invalid value (higher level error)<br>• Time-of-day frame failure<br>• Stop the time stamp function<br>• Message loss on IM (buffer overflow)<br>• Loss of information with redundancy | 16#40 |
| Invalid value | 16##00 |

## Addressing

For general information, see also Addressing (Page 399)

The logical address of the IM obtained in HW Config (corresponds to the diagnostic address, or the input address of the IM for a DP master system in DPV1 mode) is entered at the driver's block input (LADDR) by the "Generate module driver" CFC function. Any change to the LADDR block input will initiate a single check of the logical address according to the startup characteristics of the block.

Process signals that require a time stamp and are detected via an IM must be configured accordingly in HW Config.

## Error handling

I/O access error:

| | |
|---|---|
| QPERAF | The block could not access the IM. A data record could not be read. |

Block processing error:

| | |
|---|---|
| QERR | A block execution error has occurred. |

Module parameter assignment error:

| | |
|---|---|
| QPARF | Faulty block configuration: Incorrect logical basic address entered. |

Parameter assignment error:

| | |
|---|---|
| QBPARF | Faulty block configuration: The slot/channel number of an IM message does not match any slot/channel number of the block input parameters. |

Rack error:

| | |
|---|---|
| QRACKF | Failure of the rack containing the IM or IM failure. |

## IM Startup Characteristics

During startup/restart of the IM, the system will generate process interrupts once again for those records which were occupied prior to restart but had not been fetched.

The message "Startup data (incoming)" is entered as the first message of the first free data record. After restart, the system checks all monitored digital signals for changes, outputs a message if appropriate. It finally generates the message "Startup data (outgoing)".

## IMDRV_TS driver block startup characteristics

Initializes the ALARM_8P blocks using the data stored prior to the CPU transition to STOP. During an initial startup, the signal state is set to "Zero".

During startup, the block verifies the existence of an IM at its logical address set in LADDR. If this does not exist, it sets the QPARF output to TRUE, and does not access any I/Os in its subsequent cycles. QPARF = FALSE and I/O access is enabled only after the correct module has been inserted, or the logical address has been set correctly. Stored process interrupt data which were not computed before the restart will be deleted.

## Redundancy

Time stamping in H systems equipped with two IM units is redundant under the following conditions:

- Both IM units communicate via the communication (K) bus.
- No error has occurred during the update of the active and passive IM.

The SUBNET and RACK blocks report loss of redundancy (failure of an IM), separately from the IMDRV_TS block.

Time stamping is interrupted for the duration of the changeover between the active and passive IM. This period of interruption is indicated by the message "Redundant changeover" (incoming/outgoing state).

The active IM usually reports the current I/O status to the redundant IM. If this communication is disrupted, the message "Loss of information with redundancy" (entering state) is output. When the I/O statuses of the active and redundant IM are synchronized, the message "Loss of information with redundancy" (outgoing) is output.

## Time response

Not available

## Message response

The block signals system messages from the IM via the ALARM_8P block. The time-stamped hardware interrupts are forwarded to the MSG_TS IM message block via OR_32_TS.

## Operator control and monitoring:

The block does not have a faceplate.

## Additional information

You will find more information in:

Message texts of IMDRV_TS (Page 96)

### 3.13.2 I/Os of IMDRV_TS

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EV_ID | Message number for ALARM_8P | DWORD | 0 |
| EV_ID_00 | Message number 0 for ALARM_8P | DWORD | 0 |
| **LADDR** | Logic address IM | INT | 0 |
| S_CH_xxx | Slot/channel number (xxx = 000 - 127) | WORD | 0 |
| **TRIG_INF** | Edge evaluation:<br>0 = trigger 0 -> 1, 1 = trigger 1 -> 0 | STRUCT | |

#### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| M_ACK_00 | Message acknowledgment | WORD | 0 |
| Q_ERR_00 | Message error | BOOL | 0 |
| QERR | 1 = processing error block | BOOL | 1 |
| **QPARF** | 1 = parameter assignment error module | BOOL | 0 |
| **QBPARF** | 1 = parameter assignment error block | BOOL | 0 |
| **QPERAF** | 1 = I/O access error | BOOL | 0 |
| **QRACKF** | 1 = rack error | BOOL | 0 |
| **QTS_NCON** | 1 = TS_xxx is not interconnected | BOOL | 0 |
| STAT_00 | Status output | WORD | 0 |

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| TS_xxx | Time stamp (xxx = 000 – 127)<br>Bit 0: Message signal state (MsgSig)<br>Bit 1: Edge change information (TriInf)<br>Bit 2: Handshake (HdSh)<br>Byte 1:<br>Quality code of the time stamp (ST)<br>DWORD TS0:<br>Date/time stamp in ISP format (seconds)<br>DWORD TS1:<br>Date/time stamp in ISP format (fractions of seconds) | STRUCT | |
| TS_C_xxx | TS communication (xxx = 000 - 127)<br>Bit 0: Acknowledgment of transfer (HS)<br>Bit 1: Interconnection check (LI) | BYTE | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| RAC_DIAG | Rack status information | STRUCT | |

## Additional information

For additional information, refer to the sections:

Message texts of IMDRV_TS (Page 96)

### 3.13.3 Message texts of IMDRV_TS

#### Assignment of message text and message class

You will find more information in Message classes (Page 401)

| Message block | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID (ALARM_8P) | | | |
| | 1 | IM @1%d@@2%d@: Parameter assignment error Slot=@3%d@ Channel=@4%d@ | S |
| | 2 | IM @1%d@@2%d@: I/O access error: Ret_Val@5%d@ | S |
| | 3 | IM @1%d@@2%d@: Parameter assignment error LADDR | S |
| | 4 | IM @1%d@@2%d@: Output TS_xxx of S_CHxx: Slot=@3%d@ Channel=@4%d@ is not interconnected | S |
| | 5 | Reserve5 | No message |
| | 6 | Reserve6 | No message |
| | 7 | Reserve7 | No message |
| | 8 | Reserve8 | No message |
| EV_ID_00 (ALARM_8P) | | | |
| | 1 | IM @3%d@@4%d@: Startup data | S |
| | 2 | IM @3%d@@4%d@: Time-of-day frame failure | S |
| | 3 | | No message |
| | 4 | IM @3%d@@4%d@: Time difference between the frame and the internal clock may cause inaccuracy | S |
| | 5 | IM @3%d@@4%d@: Stop the time stamp function | S |
| | 6 | IM @3%d@@4%d@: Message loss on IM (buffer overflow) | S |
| | 7 | IM @3%d@@4%d@: Redundant changeover | S |
| | 8 | IM @3%d@@4%d@: Loss of information with redundancy | S |

# 3.14 MOD_1: Monitoring up to 16 channels on S7-300/400 SM modules without diagnostic capability

## 3.14.1 Description of MOD_1

### Object name (type + number)

FB 91

- MOD_1 block I/Os (Page 101)

### Area of application

The MOD_1 block monitors up to 16 channels on S7-300/400 SM modules without diagnostic capability (no mixed modules). H systems support only the modules installed in switched racks.

The block can also be used to monitor the FM 350 counter module.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The MOD_1 block is installed in its runtime group downstream of the RACK block runtime group in the above-mentioned OBs.
- The MODE_xx inputs (mode of module channels xx) are configured.
- The logical base address of the LADDR module is configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of MOD_1.
- The EN input is interconnected with the output of an AND block.
  Its inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block and EN_Mxx (xx = module number) of the RACK block.

- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function

The MOD_1 block analyzes all events that affect a module and its channels acyclically. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P reports the events.

The block is enabled by the higher-level RACK block at runtime. The event to be evaluated can be found in the CPU_DIAG start information of the OB_DIAG block. There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  - Rack failure (OB 86) (output parameter QRACKF = TRUE)
- Events that are evaluated by the MOD block:
  - Program execution error (OB 85) (output parameter QPERAF = TRUE)
  - Module removed (OB 83) (output parameter QMODF = TRUE)

"Module removed" and "I/O access error" events are reported to the OS by ALARM_8P. The diagnostics interrupt function distinguishes between module and channel errors, whereby each channel is assigned a message ID.

The system verifies during startup that the module is available (plugged in). The module status information that is read here is then available in the form of service output parameters (MOD_INF).

You will find additional information about faults in the *System Software for S7-300/400 System and Standard Functions* reference manual.

## Redundancy

The higher-level RACK block monitors the redundancy of DP master systems operating in an H system.

## MODE setting

For additional information, refer to the section "MODE settings (Page 387)".

---

**Note**

If you make changes to the MODE_xx input configurations at runtime, they are not accepted at the outputs until you set the ACC_MODE input to 1.

---

## OMODE Structure

You will find additional information in the "OMODE (Page 396)" section.

## Valid channel display

The available channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel is 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find additional information in the "Addressing (Page 399)" section.

## Error handling

The plausibility of input parameters is not checked.

You will find additional information about error handling in the "Error information of output parameter MSG_STAT (Page 398)" section.

## Service Information

To analyze faults, the module status information entered during startup is read via the MOD_INF structured output parameter. You will find additional information in the *System Software for S7-300/400 System and Standard Functions; System Status List, Module Status Information* reference manual.

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

### Time response

Not available

### Message response

MOD_1 uses ALARM_8P to report module errors. The DELAY1 and DELAY2 inputs are used to delay the output of I/O access error messages. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB 85) before it outputs the message. The message is output only under the condition that no higher-priority error is reported within this delay time. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default.
The message function can be disabled by setting EN_MSG = FALSE.

### Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

### Additional information

For additional information, refer to the sections:

Message texts and associated values of MOD_1 (Page 103)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

### 3.14.2 I/Os of MOD_1/MOD_2

The MOD_1 and MOD_2 block I/Os are identical with the exception of the number of MODE_xx and OMODE_xx. The number of monitored channels determines the corresponding number of I/O parameters (xx).

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| MODE_xx | Channel mode (xx = 00 - 15/00 - 31) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

#### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel mode (xx = 00 - 15/00 - 31) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack/station error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_1/MOD_2 (Page 103)

Maintenance status of MS (Page 403)

## 3.14.3 Message texts and associated values of MOD_1/MOD_2/MOD_3/MOD_64

### Assignment of message text and message class to the block parameters of MOD_1/MOD_2/MOD_3/MOD_64

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID | 1 | | Module @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | Module @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | QMODF | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_n_TXT@ (n = 1, 2, 3 or 64) | S |
| | 4 | QMODF | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_n_TXT@ (n = 1, 2, 3 or 64) | S |

### Assignment of associated values to the block parameters of MOD_1/MOD_2/MOD_3

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number from MOD_n_TXT (n = 1, 2, 3 or 64) (Message 1) |

You will find the message texts and their text numbers in the "Text library for MOD_1, MOD_2, MOD_3, MOD_64 (Page 417)" section.

### See also

Message Classes (Page 401)

# 3.15 MOD_2: Monitoring 32 channels on S7-300/400 SM modules without diagnostic capability

## 3.15.1 Description of MOD_2

### Object name (type + number)

FB 92

- MOD_2 block I/Os (Page 108)

### Area of application

The MOD_2 block monitors 32 channels on S7-300/400 SM modules without diagnostic capability (no mixed modules). H systems support only the modules installed in switched racks.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The MOD_2 block is installed in its runtime group downstream of the RACK block runtime group in the above-mentioned OBs.
- The MODE_xx inputs (mode of module channels xx) are configured.
- The logical base address of the LADDR module is configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of MOD_2.
- The EN input is interconnected with the output of an AND block.
  Its inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block and EN_Mxx (xx = module number) of the RACK block.

- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function

Block MOD_2 analyzes all events affecting a module and its channels in non-cyclic mode. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P reports the events.

The block is enabled to run by the higher-level RACK block. The event to be evaluated can be found in the CPU_DIAG start information of the OB_DIAG block. There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  - Rack failure (OB 86) (output parameter QRACKF = TRUE)
- Events that are evaluated by the MOD block:
  - Program execution error (OB 85) (output parameter QPERAF = TRUE)
  - Module removed (OB 83) (output parameter QMODF = TRUE)

"Module removed" and "I/O access error" events are reported to the OS by ALARM_8P. The diagnostics interrupt function distinguishes between module and channel errors, whereby each channel is assigned a message ID.

The system verifies during startup that the module is available (plugged in). The module status information that is read here is then available in the form of service output parameters (MOD_INF).

You can find additional information about errors in the *System Software for S7-300/400; System and Standard Functions* reference manual.

## Redundancy

The higher-level RACK block monitors the redundancy of DP master systems operating in an H system.

## MODE setting

You will find more information about this in the "MODE settings (Page 387)" section.

---

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

---

## OMODE Structure

You will find additional information in the "OMODE (Page 396)" section.

## Valid channel display

The available channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find additional information in the "Addressing (Page 399)" section.

## Error handling

The plausibility of input parameters is not checked.

You will find additional information about error handling in the "Error information of output parameter MSG_STAT (Page 398)" section.

## Service Information

To analyze faults, the module status information entered during startup is read via the MOD_INF structured output parameter. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; System Status List, Module Status Information*.

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

MOD_2 uses ALARM_8P to report module errors. The inputs DELAY1 and DELAY2 are used to delay the output of I/O access error messages. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB 85) before it outputs the message. The message is output only under the condition that no higher-priority error is reported within this delay time. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default.
The message function can be disabled by setting EN_MSG = FALSE.

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the sections:

Message texts and associated values of MOD_2 (Page 110)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

### 3.15.2 I/Os of MOD_1/MOD_2

The MOD_1 and MOD_2 block I/Os are identical with the exception of the number of MODE_xx and OMODE_xx. The number of monitored channels determines the corresponding number of I/O parameters (xx).

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| MODE_xx | Channel mode (xx = 00 - 15/00 - 31) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

#### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#0 0000 000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel mode (xx = 00 - 15/00 - 31) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack/station error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_1/MOD_2 (Page 110)

Maintenance status of MS (Page 403)

### 3.15.3 Message texts and associated values of MOD_1/MOD_2/MOD_3/MOD_64

#### Assignment of message text and message class to the block parameters of MOD_1/MOD_2/MOD_3/MOD_64

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID | 1 | | Module @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | Module @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | QMODF | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_n_TXT@ (n = 1, 2, 3 or 64) | S |
| | 4 | QMODF | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_n_TXT@ (n = 1, 2, 3 or 64) | S |

#### Assignment of associated values to the block parameters of MOD_1/MOD_2/MOD_3

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number from MOD_n_TXT (n = 1, 2, 3 or 64) (Message 1) |

You will find the message texts and their text numbers in the "Text library for MOD_1, MOD_2, MOD_3, MOD_64 (Page 417)" section.

#### See also

Message Classes (Page 401)

# 3.16 MOD_3: Monitoring up to 16 channels on S7-200/300/400 SM modules without diagnostic capability

## 3.16.1 Description of MOD_3

### Object name (type + number)

FB 95

- MOD_3 block I/Os (Page 115)

### Area of application

The MOD_3 block monitors up to 16 channels on S7-300/400 SM mixed modules without diagnostic capability (I/O modules). H systems support only the modules installed in switched racks.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The MOD_3 block is installed in its runtime group downstream of the RACK block runtime group in the above-mentioned OBs.
- The MODE_xx inputs are configured.
- The logical addresses LADDR and LADDR1 are configured.
- The structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the MOD_3 structures of the same name.
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function

The MOD_3 block cyclically analyzes all events that affect a module. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P reports the events. The message function can be disabled.

The block is enabled to run by the higher-level RACK block. The diagnostic event is entered in the CPU_DIAG start information of the OB_BEGIN block.

There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. MODE_00 to MODE_15 inputs are available for encoding up to 16 input channels, and MODE_16 … MODE_31 for encoding up to 16 output channels.
The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  - Rack failure (OB 86) (output parameter QRACKF = TRUE)
  - Program execution error (OB 85) (output parameter QRACKF = TRUE)
- Events that are evaluated by the MOD block:
  - I/O access error (OB 85) (output parameter QPERAF = TRUE)
  - Module removed (OB 83) (output parameter QMODF = TRUE)

"Module removed" and "I/O access error" events are reported to the OS by ALARM_8P. The diagnostics interrupt function distinguishes between module and channel errors, whereby each channel is assigned a message ID.

The system verifies during startup that the module is available (plugged in). The module status information read here makes this data available in the form of service output parameters (MOD_INF).

You will find additional information about faults in the *System Software for S7-300/400 System and Standard Functions* reference manual.

## Redundancy

Block MOD_3 supports segment redundancy of H systems operating with distributed I/Os. If you want to use this function, you must configure the SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs of the SUBNET block with the numbers of the redundant segments. If there is no segment redundancy, the remaining input must be set to the (default) value 16#FF.

## MODE setting

You will find more information about this in the "MODE settings (Page 387)" section.

---

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

---

## OMODE Structure

You will find additional information in the "OMODE (Page 396)" section.

## Valid channel display

The available channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find additional information about this in the "Addressing (Page 399)" section.

## Error handling

The plausibility of input parameters is not checked.

You will find additional information about error handling in the "Error information of output parameter MSG_STAT (Page 398)" section.

## Service Information

To analyze faults, the module status information entered during startup is read via the MOD_INF structured output parameter. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; System Status List, Module Status Information*.

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

Block MOD_3 uses ALARM_8P to report module errors. The inputs DELAY1 and DELAY2 are used to delay the output of I/O access error messages. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB 85) before it outputs the message. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default. The message function can be disabled by setting EN_MSG = FALSE.

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the sections:

Message texts and associated values of MOD_3 (Page 117)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.16.2 I/Os of MOD_3

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **LADDR** | Logical address of the input channels | INT | 0 |
| **LADDR1** | Logical address of the output channels | INT | 0 |
| MODE_xx | Channel xx mode | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#0 0000 000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | 0 |

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel xx mode | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = failure of the rack | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Def-ault |
|---|---|---|---|
| ACC_MODE | 1 = Accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_3 (Page 117)

Maintenance status of MS (Page 403)

## 3.16.3 Message texts and associated values of MOD_1/MOD_2/MOD_3/MOD_64

### Assignment of message text and message class to the block parameters of MOD_1/MOD_2/MOD_3/MOD_64

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID | 1 | | Module @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | Module @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | QMODF | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_n_TXT@ (n = 1, 2, 3 or 64) | S |
| | 4 | QMODF | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_n_TXT@ (n = 1, 2, 3 or 64) | S |

### Assignment of associated values to the block parameters of MOD_1/MOD_2/MOD_3

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number from MOD_n_TXT (n = 1, 2, 3 or 64) (Message 1) |

You will find the message texts and their text numbers in the "Text library for MOD_1, MOD_2, MOD_3, MOD_64 (Page 417)" section.

### See also

Message Classes (Page 401)

# 3.17 MOD_4: Monitoring ET 200S modules downstream of a Y-Link

## 3.17.1 Description of MOD_4

### Object name (type + number)

FB 119

- MOD_4 block I/Os (Page 122)

### Area of application

Block MOD_4 monitors modules (up to 16 channels) of an ET 200S acting as a DPV0 slave (IM 151 High Feature) downstream of a Y link.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- Block MOD_4 is installed in its runtime group downstream of the OB_DIAG1 block, which is used for the ET 200S downstream of a Y link, in the above-mentioned OBs.
- The MODE_xx inputs (mode of module channels xx) are configured.
- The logical base address of the LADDR module is configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the OB_DIAG1 blocks (for DP slave downstream of a Y link, and an OB_DIAG1 for each Y link) are interconnected with the IN_OUT structures RAC_DIAG (DP slave) and RAC_DIAG_L (Y link) of MOD_4.
- The EN input is interconnected with the output of an AND block.
  whose inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block, EN_F of the OB_DIAG1 block for the Y link, EN_F of the OB_DIAG1 block for the ET 200S downstream of the Y link, and EN_Mxx (xx = module slot number in the ET 200S) of the DPDIAGV0 block.
- The DPA_M input is interconnected with the DPA_Mxx (xx= module slot number in the ET 200S) output of the DPDIAGV0 block.
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function

Block MOD_4 analyzes all events related to an ET 200S module acyclically. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P reports events separately for each module.

The block enabled to run by the higher-level DPDIAGV0 block. The event to be evaluated is available at input DPA_M. Possible byte assignments:

0000000 = Module OK

0000001 = Module error

0000010 = Wrong module

0000011 = Module missing

00001xx = ET 200S failure; x = irrelevant

There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This only occurs if the module status changes during startup, or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the OB_DIAG1 block:
  - Rack failure (OB 86,OB 83) (output parameter QRACKF = TRUE)
- Events that are evaluated by the MOD block:
  - Module diagnostics (OB 82) (output parameter QMODF = TRUE)

ALARM_8P is used to report "Module error ", "Wrong module " or "Module missing " events to the OS.

## Redundancy

You can not use redundant DP slaves downstream of a Y link.

## MODE setting

You will find more information about this in the "MODE settings (Page 387)" section.

---

Note

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

---

## OMODE Structure

You will find further information in the "OMODE (Page 396)" section.

## Valid channel display

The available channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find additional information about this in the "Addressing (Page 399)" section.

## Error handling

The plausibility of input parameters is not checked.

You will find further information about error handling in

"Error information of output parameter MSG_STAT (Page 398)".

## Startup characteristics

A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

MOD_4 uses ALARM_8P to report module errors. The message function can be disabled by setting EN_MSG = FALSE.

The block generates the maintenance status MS.

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For further information, refer to the sections:

Message texts and associated values of MOD_4 (Page 124)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.17.2 I/Os of MOD_4

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| **DPA_M** | Module status | BYTE | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 1 |
| **LADDR** | Logic address of the module | INT | 0 |
| MODE_xx | Mode channel (xx = 00 - 15) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **PADP_ADR** | DP address of ET 200S | BYTE | 255 |
| **RACK_NO** | Rack number (Y-Link) | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#0 0000 000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit 0= Release for maintenance<br>Bits 1-31 = Reserve | DWORD | 0 |
| **EXT_STAT** | Release for maintenance - extended status | DWORD | 0 |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Mode channel (xx = 00 - 15) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QRACKF | 1 = rack/station error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics of the DP slave downstream of Y link | STRUCT | |
| RAC_DIAG_L | System structure: Rack diagnostics for the Y link | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_4 (Page 124)

Maintenance status of MS (Page 403)

### 3.17.3 Message texts and associated values of MOD_4

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID | 1 | Module @1%d@/@2%d@/@3%d@/@4%d@: Error | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@4%d@: Wrong | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@4%d@: Missing | S |

#### Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) (Y link) |
| | 3 | PADP_ADR | DP address of ET 200S |
| | 4 | SLOT_NO | Slot number (byte) |

#### See also

Message Classes (Page 401)

# 3.18 MOD_64: Monitoring 64 channels on S7-300 SM modules without diagnostic capability

## 3.18.1 Description of MOD_64

### Object name (type + number)

FB 137

- MOD_64 block I/Os (Page 129)

### Area of application

The MOD_64 block monitors 64 channels on S7-300 SM modules without diagnostic capability (no mixed modules). H systems support only the modules installed in switched racks.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The MOD_64 block is installed in its runtime group downstream of the RACK block runtime group in the above-mentioned OBs.
- The MODE_xx inputs (mode of module channels xx) are configured.
- The logical base address of the LADDR module is configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of MOD_2.
- The EN input is interconnected with the output of an AND block.
  Its inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block and EN_Mxx (xx = module number) of the RACK block.

- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function

The MOD_64 block analyzes all events that affect a module and its channels acyclically. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P reports the events.

The block is enabled to run by the higher-level RACK block. The event to be evaluated can be found in the CPU_DIAG start information of the OB_DIAG block. There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  - Rack failure (OB 86) (output parameter QRACKF = TRUE)
- Events that are evaluated by the MOD block:
  - Program execution error (OB 85) (output parameter QPERAF = TRUE)
  - Module removed (OB 83) (output parameter QMODF = TRUE)

"Module removed" and "I/O access error" events are reported to the OS by ALARM_8P. The diagnostics interrupt function distinguishes between module and channel errors, whereby each channel is assigned a message ID.

The system verifies during startup that the module is available (plugged in). The module status information that is read here is then available in the form of service output parameters (MOD_INF).

You can find additional information about errors in the *System Software for S7-300/400; System and Standard Functions* reference manual.

## Redundancy

The higher-level RACK block monitors the redundancy of DP master systems operating in an H system.

## MODE setting

You will find more information about this in the "MODE settings (Page 387)" section.

---

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

---

## OMODE Structure

You will find additional information in the "OMODE (Page 396)" section.

## Valid channel display

The available channels on a module are displayed in the CH_EXIST or CH_EXIST_2 output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK or CH_OK_2 displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel = 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find additional information in the "Addressing (Page 399)" section.

## Error handling

The plausibility of input parameters is not checked.

You will find additional information about error handling in the "Error information of output parameter MSG_STAT (Page 398)" section.

## Service Information

To analyze faults, the module status information entered during startup is read via the MOD_INF structured output parameter. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; System Status List, Module Status Information*.

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

### Time response

Not available

### Message response

MOD_64 uses ALARM_8P to report module errors. The inputs DELAY1 and DELAY2 are used to delay the output of I/O access error messages. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB 85) before it outputs the message. The message is output only under the condition that no higher-priority error is reported within this delay time. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default.
The message function can be disabled by setting EN_MSG = FALSE.

### Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

### Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_64 (Page 131)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.18.2 I/Os of MOD_64

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| MODE_xx | Mode channel (xx = 00 - 63) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists (0 to 31) | DWORD | 0 |
| CH_EXIST_2 | Channel exists (32 to 63) | DWORD | 0 |
| CH_OK | Channel OK (0 to 31) | DWORD | 0 |
| CH_OK_2 | Channel OK (32 to 63) | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel<br>(xx = 00 - 63)<br>Bit 0= Release for maintenance<br>Bits 1-31 = Reserve | DWORD | 0 |
| **EXT_STAT** | Release for maintenance - extended status | DWORD | 0 |
| **EXT_STAT_2** | Release for maintenance - extended status 2 | DWORD | 0 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Mode channel (xx = 00 - 63) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack/station error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_64 (Page 131)

Maintenance status of MS (Page 403)

## 3.18.3 Message texts and associated values of MOD_1/MOD_2/MOD_3/MOD_64

### Assignment of message text and message class to the block parameters of MOD_1/MOD_2/MOD_3/MOD_64

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID | 1 | | Module @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | Module @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | QMODF | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_n_TXT@ (n = 1, 2, 3 or 64) | S |
| | 4 | QMODF | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_n_TXT@ (n = 1, 2, 3 or 64) | S |

### Assignment of associated values to the block parameters of MOD_1/MOD_2/MOD_3

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number from MOD_n_TXT (n = 1, 2, 3 or 64) (Message 1) |

You will find the message texts and their text numbers in the "Text library for MOD_1, MOD_2, MOD_3, MOD_64 (Page 417)" section.

### See also

Message Classes (Page 401)

# 3.19 MOD_CP: CP 341/441 diagnostics

## 3.19.1 Description of MOD_CP

### Object name (type + number)

FB 98

- MOD_CP block I/Os (Page 136)

### Area of application

Block MOD_CP monitors a CP 341 or CP 441 serial communication module. H systems only support modules in switched racks.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The MOD_CP block is installed in its runtime group downstream of the RACK block runtime group in the above-mentioned OBs.
- The SUBN1_ID, SUBN2_ID, SUBN_TYP, RACK_NO, and SLOT_NO inputs are configured.
- The logical base address LADDR of the module is configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of MOD_CP.
- The EN input is interconnected with the output of an AND block.
  Its inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block and EN_Mxx (xx = module number) of the RACK block.

## Function and method of operation

Block MOD_CP analyzes all events that affect a module and its channels acyclically. and generates a value status for the serial communication blocks (such as RCV_341). ALARM_8P is used to report these events.

The higher-level RACK block enables the MOD_CP block to run. The event to be evaluated is stored in the CPU_DIAG start and diagnostic information of the OB_BEGIN block. An input (MODE_00) is assigned to the communication channel of the module. Input MODE_01 is reserved for the second CP 441 communication channel, which can be used to report the communication-channel configuration data from HW Config. The driver generator cannot access this data at present, so the SND_341 or RCV_341 block does not evaluate the data either. If user-specific blocks are used, the user can define codes at the MODE (Page 387) input for use in these user-specific blocks. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output. This occurs only during startup or if you set ACC_MODE = TRUE. The current value status of the communication channel is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  - Rack failure (OB 86) (output parameter QRACKF = TRUE)
- Events that are evaluated by the MOD block:
  - Program execution error (OB 85) (output parameter QPERAF = TRUE)
  - Module removed (OB 83) (output parameter QMODF = TRUE)
  - Diagnostic interrupt (OB 82) Distinguishing between module errors and channel errors

## The following events are module errors (QMODF = TRUE output parameter):

ALARM_8P is used to report "Module removed", "I/O access error" and "Diagnostic interrupt" events to WinCC.

It is only with the CP441 that the system distinguishes between module and channel errors in response to a diagnostic interrupt; two message IDs (parameter assignment error, line break) are assigned to each channel.

## Redundancy

The higher-level RACK block monitors the redundancy of DP master systems operating in an H system.

## Valid channel display

The existing channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Overload behavior

The MOD_CP block counts the OB 82 calls. The counter is reset in OB1. If more than five OB 82 events occur in succession before the cycle control point is reached (OB 1), these events are rejected and the message "OB 82 DP master failure: x Rack: y Slot: z" is output.

## Time response

Not available

## Message response

MOD_CP uses ALARM_8P to report module errors. The inputs DELAY1 and DELAY2 are used to delay the output of I/O access error messages. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB 85) before it outputs the message. The message is output only under the condition that no higher-priority error is reported within this delay time. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default.
The message function can be disabled by setting EN_MSG = FALSE.

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For further information, refer to the following sections:

Message texts and associated values of MOD_CP (Page 138)

Maintenance status of MS (Page 403)

### 3.19.2 I/Os of MOD_CP

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| MODE_00 | Channel 1 mode | WORD | 0 |
| MODE_01 | Channel 2 mode (CP 441 only) | WORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

#### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DIAG_INF | System structure: Diagnostic information | STRUCT | |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_00 | Value status/channel 1 mode | DWORD | 0 |
| OMODE_01 | Value status/channel 2 mode (CP 441 only) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| **QMODF** | 1 = module removed/defective | BOOL | 0 |
| **QPERAF** | 1 = I/O access error | BOOL | 0 |
| **QRACKF** | 1 = rack error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_CP (Page 138)

Maintenance status of MS (Page 403)

### 3.19.3 Message texts and associated values of MOD_CP

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID | 1 | QMODF | CP @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | CP @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | | CP @1%d@/@2%d@/@3%d@: @4W%t#MOD_CP_TXT@ | S |
| | 4 | | CP @1%d@/@2%d@/@3%d@: Wrong parameter | S |
| | 5 | | CP @1%d@/@2%d@/@3%d@: Wire break | S |
| | 6 | | CP @1%d@/@2%d@/@3%d@/2: Wrong parameter | S |
| | 7 | | CP @1%d@/@2%d@/@3%d@/2: Wire break | S |
| | 8 | | CP @1%d@/@2%d@/@3%d@: Multiple diagnostic interrupt | S |

You will find the message texts and their text numbers in the "Text library for MOD_CP (Page 417)" section.

#### Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |

SUBN_ID = SUBN1_ID. If SUBN1_ID = 16#FF, the associated value is substituted with SUBN2_ID.

#### See also

Message Classes (Page 401)

# 3.20 MOD_D1: Monitoring up to 16 channels on S7-300/400 SM modules with diagnostic functions

## 3.20.1 Description of MOD_D1

### Object name (type + number)

FB 93

- MOD_D1 block I/Os (Page 146)

### Area of application

The MOD_D1 block can monitor the following modules:

- Power modules
- ET200S counter modules
- FM350 counter modules
- up to 16 channels on S7-300/400 SM modules with diagnostic functions (no mixed modules)
- and the power supplies of an ET 200iSP in a redundant configuration.

H systems support only the modules installed in switched racks.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The MOD_D1 block is installed in its runtime group downstream of the RACK block runtime group in the above-mentioned OBs.
- The MODE_xx (mode of the channels xx of the module), SUBN1_ID, SUBN2_ID, and SUBN_TYP inputs are configured.
- The logical base address LADDR of the module is configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of MOD_D1.
- The EN input is interconnected with the output of an AND block.
  Its inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block and EN_Mxx (xx = module number) of the RACK block.
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function and method of operation

Block MOD_D1 analyzes all events that affect a module and its channels acyclically. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P is used to report these events.

MOD_D1 monitors the redundant power supplies for ET 200iSP and reports the failure of a power supply via the slot number of the interface module. The maintenance status (MS) of the module shows, if the power supply fails "Maintenance: Requirement Moderate" an.

Block MOD_D1 is enabled by the higher-level RACK block at runtime. The event to be evaluated is stored in the CPU_DIAG start and diagnostic information of the OB_BEGIN block. There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  – Rack failure (OB 86) (output parameter QPERAF = TRUE)
- Events that are evaluated by the MOD block:
  – Program execution error (OB 85) (output parameter QPERAF = TRUE)
  – Module removed (OB 83) (output parameter QMODF = TRUE)
  – Diagnostic interrupt (OB 82) Distinguishing between module errors and channel errors

### The following events are module errors (QMODF = TRUE output parameter):

- External auxiliary voltage missing
- Front connector missing
- Module not configured
- Wrong parameter in module
- Wrong/missing module
- Communication error at a CPU module
- Timeout (watchdog)
- Loss of internal power at a module
- Rack failure
- CPU failure
- EPROM error
- RAM error
- ADC/DAC error
- Fuse tripped
- Power supply 1: Error
- Power supply 2: Error

### The following events are channel errors (value status "invalid value", OMODE_xx = 16#00xxxx):

- Configuration/parameter assignment error
- Common-mode errors (analog I/Os only)
- Short-circuit to P
- Short circuit to M
- Interruption at the output transistor
- Wire break
- Reference channel error (analog inputs only)
- Measuring range underflow (only analog inputs)
- Measuring range overflow (analog inputs only)
- Missing load voltage (only analog and digital outputs)
- Missing sensor power supply (digital outputs only)
- Fuse tripped (only digital outputs)
- Mass error (digital I/Os only)
- Excess temperature (only digital outputs)

ALARM_8P is used to report "Module removed", "I/O access error", and "Diagnostic interrupt" events to WinCC.

The diagnostics interrupt function distinguishes between module and channel errors, whereby each channel is assigned a message ID. Only one incoming/outgoing event can be reported for each channel. As long as an incoming message is queued at a channel, further messages on new events at this channel will be lost.

If the event is defined uniquely in the diagnostic information, the corresponding text will be entered in the message. If there are ambiguous entries, the text of the first set bit in the error byte of the diagnostic information will be displayed. When using modules assigned diagnostic functions and more than one error byte for diagnostic information, only the channel xx error text will be output if the error information is not displayed in the first error byte.

The system verifies during startup that the module is available (plugged in). The module status information that is read here is then available in the form of service output parameters (MOD_INF).

Detailed information about the errors is entered in the DIAG_INF output parameter of data type STRUCT. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; Diagnostic Data, Byte 0 to Byte 8, Structure of Channel-Specific Diagnostic Data*.

---

Note

If you run a HART module in HART MODE (Page 387) =16#070C, any HART protocol errors/configuration changes will be masked by the MOD_D1 driver block, and will not be signaled as channel errors.

---

## Redundancy

The higher-level RACK block monitors the redundancy of DP master systems operating in an H system.

## MODE setting

You will find more information about this in the "MODE settings (Page 387)" section.

---

Note

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

---

## OMODE Structure

You will find further information in the "OMODE (Page 396)" section.

## Valid channel display

The available channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find more information about this in "Addressing (Page 399)".

HART modules with read/write access to the process image are configured in the same way as input modules. The set I/O range must always be identical.

Example: SM332 AO 2x0/4..20mA HART 332-5TB00-0AB0:

| Address input range (HW Config) | Address output range (HW Config) | LADDR (decimal/hex) |
|---|---|---|
| 544 | 544 | 544 / 16#0220 |

## Error handling

The plausibility of input parameters is not checked.
You will find further information about error handling in the "Error information of output parameter MSG_STAT (Page 398)" section.

## Service Information

To analyze faults, the module status information entered during startup is read via the MOD_INF structured output parameter. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; System Status List, Module Status Information*.

Following a diagnostic interrupt, you will also find detailed module diagnostic information in the MODDIAG0 to MODDIAG8 output parameters. You will find more information in the reference manual *System Software for S7-300/400 System and Standard Functions; Diagnostic Data, Byte 0 to Byte 8*.

The CHDIAG00 to CHDIAG15 output parameters contain detailed channel-status information. You will find more information in the reference manual *System Software for S7-300/400 System and Standard Functions; Structure of Channel-Specific Diagnostic Data*.

The system resets this diagnostic information after a diagnostic interrupt has been reported outgoing (no further channel or module errors are queued).

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Overload behavior

The MOD_D1 block counts the OB82 calls. The counter is reset in OB1. If more than two OB82 events occur in succession before the cycle control point is reached (OB1), these events are rejected, and the message "OB82 DP master failure: x Rack: y Slot: z" is output.

## Time response

Not available

## Message response

MOD_D1 uses ALARM_8P_1 to report module errors. The function also calls ALARM_8P_2 and ALARM_8P_3 which are intended for channel errors. The DELAY1 and DELAY2 inputs are used to delay the output of I/O access error messages. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB85) before it outputs the message. The message is output only under the condition that no higher-priority error is reported within this delay time. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default.
The message function can be disabled by setting EN_MSG = FALSE.

## Flutter suppression

The "Flutter suppression" function is used to delay the outgoing of a message by a configurable period.

The flutter time is entered at the channel block at the FlutTmIn parameter. The high byte of the DXCHG parameter of the channel blocks contains the flutter time.

Flutter suppression comes into effect when FlutEN = 1 or FlutTmIn > 0 is set at the channel block.

There is only one flutter message per module. The delay times and fault messages are channel-specific. The fault messages are extended by at least the delay time. Flutter exists if the fault messages "Outgoing" and then "Incoming" are present within the delay time.

The last fluttering channel and its set delay time deactivates the flutter message.

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_D1 (Page 148)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.20.2 I/Os of MOD_D1/MOD_D2

The MOD_D1 and MOD_D2 block I/Os are identical, with the exception of the number of MODE_xx and OMODE_xx. The number of monitored channels determines the corresponding number of I/O parameters (xx).

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number (x = 1 - 3) | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| MODE_xx | Operating mode channel xx (xx = 00 - 15 / 00 - 31) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | 0 |
| DIAG_INF | System structure: Diagnostic information | STRUCT | 0 |

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression | DWORD | 16#000 00000 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACKx | Message acknowledgment (x = 1 - 3) | WORD | 0 |
| MSGSTATx | Message error information (x = 1 - 3) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel mode (xx = 00 - 15/00 - 31) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_D1 (Page 148)

Message texts and associated values of MOD_D2 (Page 158)

Maintenance status of MS (Page 403)

## 3.20.3 Message texts and associated values of MOD_D1

### Assignment of message text and message class

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 | 1 | QMODF | Module @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | Module @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | QMODF | Module @1%d@/@2%d@/@3%d@: @5W%t#MOD_D1_TXT@ | S |
| | 4 | | Module @1%d@/@2%d@/@3%d@: Multiple diagnostic interrupt | S |
| | 5 | | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_D1_TXT@ | S |
| | 6 | | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_D1_TXT@ | S |
| | 7 | | Module @1%d@/@2%d@/@3%d@: @5W%t#MOD_D1_TXT@ | F |
| | 8 | | - | |
| EV_ID2 | 1 | - | Module @1%d@/@2%d@/@3%d@: Error channel 00 @4W%t#MOD_D1_TXT@ | S |
| | ... ... | | ... ... | ... ... |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error channel 07 @4W%t#MOD_D1_TXT@ | S |
| | | | | |
| EV_ID3 | 1 | - | Module @1%d@/@2%d@/@3%d@: Error channel 08 @4W%t#MOD_D1_TXT@ | S |
| | ... ... | | ... ... | ... ... |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error channel 15 @4W%t#MOD_D1_TXT@ | S |

You will find the message texts and their text numbers in "Text library for MOD_D1 (Page 419)".

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID1 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number (message 5) from MOD_D1_TXT |
| | 5 | | Text number (message 3) from MOD_D1_TXT |
| | | | |
| EV_ID2 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number (messages 1 - 8) from MOD_D1_TXT |
| | | | |
| EV_ID3 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number (messages 1 - 8) from MOD_D1_TXT |

## See also

Message Classes (Page 401)

# 3.21 MOD_D2: Monitoring up to 32 channels of S7-300/400 SM modules with diagnostic functions

## 3.21.1 Description of MOD_D2

### Object name (type + number)

FB 94

- MOD_D2 block I/Os (Page 156)

### Area of application

Block MOD_D2 monitors up to 32 channels on S7-300/400 SM modules with diagnostic capability (no mixed modules). H systems support only the modules installed in switched racks.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The MOD_D2 block is installed in its runtime group downstream of the RACK block runtime group in the above-mentioned OBs.
- The MODE_xx (mode of the channels xx of the module), SUBN1_ID, SUBN2_ID, and SUBN_TYP inputs are configured.
- The logical base address LADDR of the module is configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of MOD_D2.
- The EN input is interconnected with the output of an AND block.
  Its inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block and EN_Mxx (xx = module number) of the RACK block.

- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function and method of operation

Block MOD_D2 analyzes all events that affect a module and its channels acyclically. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P is used to report these events. The message function can be disabled.

The block is enabled by the higher-level RACK block at runtime. The event to be evaluated is stored in the CPU_DIAG start and diagnostic information of the OB_BEGIN block. There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  - Rack failure (OB 86) (output parameter QRACKF = TRUE)
- Events that are evaluated by the MOD block:
  - Program execution error (OB 85) (output parameter QPERAF = TRUE)
  - Module removed (OB 83) (output parameter QMODF = TRUE)
  - Diagnostic interrupt (OB 82) Distinguishing between module errors and channel errors

## The following events are module errors (QMODF = TRUE output parameter):

- External auxiliary voltage missing
- Front connector missing
- Module not configured
- Wrong parameter in module
- Wrong/missing module
- Communication error at a CPU module
- Timeout (watchdog)
- Loss of internal power at a module
- Rack failure
- CPU failure
- EPROM error
- RAM error

- ADC/DAC error
- Fuse tripped

### The following events are channel errors (value status "invalid value", OMODE_xx = 16#00xxxx):

- Configuration/parameter assignment error
- Common-mode errors (analog I/Os only)
- Short-circuit to P
- Short circuit to M
- Interruption at the output transistor
- Wire break
- Reference channel error (analog inputs only)
- Measuring range underflow (only analog inputs)
- Measuring range overflow (analog inputs only)
- Missing load voltage (only analog and digital outputs)
- Missing sensor power supply (digital outputs only)
- Fuse tripped (only digital outputs)
- Mass error (digital I/Os only)
- Excess temperature (only digital outputs)

ALARM_8P is used to report "Module removed", "I/O access error", and "Diagnostic interrupt" events to WinCC.

The diagnostics interrupt function distinguishes between module and channel errors, whereby each channel is assigned a message ID. Only one incoming or outgoing event can be reported for each channel. As long as an incoming message is queued at a channel, further messages on new events at this channel will be lost.

If the event is defined uniquely in the diagnostic information, the corresponding text will be entered in the message. If ambiguous entries exist, the text of the first set bit in the error byte of the diagnostic information will be displayed. When using modules assigned diagnostic functions and more than one error byte for diagnostic information, only the channel xx error text will be output if the error information is not displayed in the first error byte.

The system verifies during startup that the module is available (plugged in). The module status information read here makes this data available in the form of service output parameters (MOD_INF).

Detailed information about the errors is entered in the DIAG_INF output parameter of data type STRUCT. You will find further information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; Diagnostic Data, Byte 0 to Byte 8, Structure of Channel-Specific Diagnostic Data*.

## Redundancy

The block supports segment redundancy of CPU 417H for distributed I/Os. The SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs are configured with the numbers of the redundant segments. If there is no segment redundancy, the remaining input must be set to the (default) value 16#FF.

## MODE setting

You will find more information about this in the "MODE settings (Page 387)" section.

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

## OMODE Structure

You will find further information in the "OMODE (Page 396)" section.

## Valid channel display

The available channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find additional information about this in the "Addressing (Page 399)".

## Error handling

The plausibility of input parameters is not checked.
You will find further information about error handling in "Error information of output parameter MSG_STAT (Page 398)".

## Service Information

To analyze faults, the module status information entered during startup is read via the MOD_INF structured output parameter. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; System Status List, Module Status Information*.

Following a diagnostic interrupt, you will also find detailed module diagnostic information in the MODDIAG0 to MODDIAG10 output parameters. You will find more information in the reference manual *System Software for S7-300/400 System and Standard Functions; Diagnostic Data, Byte 0 to Byte 10*.

The CHDIAG00 to CHDIAG31 output parameters contain detailed channel status information. You will find more information in the reference manual *System Software for S7-300/400 System and Standard Functions; Structure of Channel-Specific Diagnostic Data*.

The system resets this diagnostic information after a diagnostic interrupt has been reported outgoing (no further channel or module errors are queued).

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

MOD_D2 uses ALARM_8P_1 to report module errors. In addition, the error blocks ALARM_8P_2, ALARM_8P_3, ALARM_8P_4 and ALARM_8P_5 are called. The inputs DELAY1 and DELAY2 are used to delay the output of I/O access error messages. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB 85) before it outputs the message. The message is output only under the condition that no higher-priority error is reported within this delay time. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default.
The message function can be disabled by setting EN_MSG = FALSE.

## Flutter suppression

The "Flutter suppression" function is used to delay the outgoing of a message by a configurable period.

The flutter time is entered at the channel block at the FlutTmIn parameter. The high byte of the DXCHG parameter of the channel blocks contains the flutter time.

Flutter suppression comes into effect when FlutEN = 1 or FlutTmIn > 0 is set at the channel block.

There is only one flutter message per module. The delay times and fault messages are channel-specific. The fault messages are extended by at least the delay time. Flutter exists if the fault messages "Outgoing" and then "Incoming" are present within the delay time.

The last fluttering channel and its set delay time deactivates the flutter message.

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For further information, refer to the following sections:

Message texts and associated values of MOD_D2 (Page 158)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

### 3.21.2 I/Os of MOD_D1/MOD_D2

The MOD_D1 and MOD_D2 block I/Os are identical, with the exception of the number of MODE_xx and OMODE_xx. The number of monitored channels determines the corresponding number of I/O parameters (xx).

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number (x = 1 - 3) | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| MODE_xx | Operating mode channel xx (xx = 00 - 15 / 00 - 31) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

#### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#000 00000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | 0 |
| DIAG_INF | System structure: Diagnostic information | STRUCT | 0 |

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression | DWORD | 16#000 00000 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACKx | Message acknowledgment (x = 1 - 3) | WORD | 0 |
| MSGSTATx | Message error information (x = 1 - 3) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel mode (xx = 00 - 15/00 - 31) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_D1 (Page 148)

Message texts and associated values of MOD_D2 (Page 158)

Maintenance status of MS (Page 403)

## See also

MODE settings for SM modules (Page 387)

OMODE settings for SM modules (Page 396)

### 3.21.3 Message texts and associated values of MOD_D2

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 | 1 | QMODF | Module @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | Module @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | QMODF | Module @1%d@/@2%d@/@3%d@: @5W%t#MOD_D2_TXT@ | S |
| | 4 | - | Module @1%d@/@2%d@/@3%d@: Multiple diagnostic interrupt | S |
| | 5 | - | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_D2_TXT@ | S |
| | 6 | | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_D1_TXT@ | S |
| | 7 | - | | S |
| | 8 | - | | S |
| EV_ID2 | 1 | - | Module @1%d@/@2%d@/@3%d@: Error channel 00 @4W%t#MOD_D2_TXT@ | S |
| | ... | | ... | |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error channel 07 @4W%t#MOD_D2_TXT@ | S |
| EV_ID3 | 1 | - | Module @1%d@/@2%d@/@3%d@: Error channel 08 @4W%t#MOD_D2_TXT@ | S |
| | ... | | ... | |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error channel 15 @4W%t#MOD_D2_TXT@ | S |
| EV_ID4 | 1 | - | Module @1%d@/@2%d@/@3%d@: Error channel 16 @4W%t#MOD_D2_TXT@ | S |
| | ... | | ... | |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error channel 23 @4W%t#MOD_D2_TXT@ | S |

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID5 | 1 | - | Module @1%d@/@2%d@/@3%d@: Error channel 24 @4W%t#MOD_D2_TXT@ | S |
| | ... | | ... | |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error channel 31 @4W%t#MOD_D2_TXT@ | S |

You will find the message texts and their text numbers in "Text library for MOD_D2 (Page 417)".

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID1 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | - | Text number (messages 1 - 3) of MOD_D2_TXT |
| | | | |
| EV_ID2 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | - | Text number (message 5) of MOD_D2_TXT |
| | 5 | - | Text number (message 3) of MOD_D2_TXT |
| | | | |
| EV_ID3 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | - | Text number (messages 1 - 8) of MOD_D2_TXT |
| | | | |
| EV_ID4 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | - | Text number (messages 1 - 8) of MOD_D2_TXT |
| | | | |

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID5 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | - | Text number (messages 1 - 8) of MOD_D2_TXT |

## See also

Message Classes (Page 401)

# 3.22 MOD_D3: Monitoring of hybrid modules with diagnostic capability

## 3.22.1 Description of MOD_D3

### Object name (type + number)

FB 134

- MOD_D3 block I/Os (Page 169)

### Area of application

Block MOD_D3 monitors a maximum of up to 16 channels on S7-300 SM modules with diagnostics functions. H systems support only the modules installed in switched racks.

MOD_D3 includes all the functionality of MOD_D1, plus additional functions for diagnostic evaluation of multiple channel types in a diagnostic data record. The block also fully supports 4-byte channel-specific diagnostics.
Note: MOD_D1 only evaluated 8 selected bits of the 4-byte channel-specific diagnosis.

The modules supported are the ET 200PRO modules:
6ES7 148 4FC00 0AB0 -> 8DI/4DO
6ES7 148 4FA00 0AB0 -> 8/16 DI

and the ET 200M HART modules:
6ES7 331-7TF01-0AB0 -> AI8 HART
6ES7 332-8TF01-0AB0 -> AO8 HART

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The MOD_D3 block is installed in its runtime group downstream of the RACK-block runtime group in the above-mentioned OBs.
- The MODE_xx (mode of the channels xx of the module), SUBN1_ID, SUBN2_ID, and SUBN_TYP inputs are configured.
- The logical base address LADDR of the module is configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of MOD_D3.
- The EN input is interconnected with the output of an AND block.
  Its inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block and EN_Mxx (xx = module number) of the RACK block.
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function and method of operation

The MOD_D3 block analyzes all events affecting a module and its channels acyclically. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P is used to report these events.

Execution of the MOD_D3 block is enabled by the higher-level RACK block. The event to be evaluated is stored in the CPU_DIAG start and diagnostic information of the OB_BEGIN block. There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  - Rack failure (OB 86) (output parameter QPERAF = TRUE)
- Events that are evaluated by the MOD block:
  - Program execution error (OB 85) (output parameter QPERAF = TRUE)
  - Module removed (OB 83) (output parameter QMODF = TRUE)
  - Diagnostic interrupt (OB 82) Distinguishing between module errors and channel errors

### The following events are module errors (QMODF = TRUE output parameter):

- External auxiliary voltage missing
- Front connector missing
- Module not configured
- Wrong parameter in module
- Wrong/missing module
- Communication error at a CPU module
- Timeout (watchdog)
- Loss of internal power at a module
- Rack failure
- CPU failure
- EPROM error
- RAM error
- ADC/DAC error
- Fuse tripped
- Power supply 1: Error
- Power supply 2: Error

### The following events are channel errors (value status "invalid value", OMODE_xx = 16#00xxxx):

- Configuration/parameter assignment error
- Common-mode errors (analog I/Os only)
- Short-circuit to P
- Short circuit to M
- Interruption at the output transistor
- Wire break
- Reference channel error (analog inputs only)
- Measuring range underflow (only analog inputs)
- Measuring range overflow (only analog inputs)
- Missing load voltage (only analog and digital outputs)
- Missing sensor power supply (digital outputs only)
- Fuse tripped (only digital outputs)
- Mass error (only digital I/Os)
- Excess temperature (only digital outputs)
- Undervoltage

- Overvoltage
- Overload
- Hardware interrupt
- Actuator warning
- Safety shutdown
- Ambiguous error
- Error 1 in actuator/sensor
- Error 2 in actuator/sensor
- Channel temporarily not available

ALARM_8P is used to report "Module removed", "I/O access error" and "Diagnostics interrupt" events to the OS.

The diagnostics interrupt function distinguishes between module and channel errors, whereby each channel is assigned a message ID. Only one incoming/outgoing event can be reported for each channel. As long as an incoming message is queued at a channel, further messages on new events at this channel will be lost.

If the event is defined uniquely in the diagnostic information, the corresponding text will be entered in the message. If ambiguous entries exist, the text of the first set bit in the error byte of the diagnostic information will be displayed. When using modules assigned diagnostic functions and more than one error byte for diagnostic information, only the channel xx error text will be output if the error information is not displayed in the first error byte.

The system verifies during startup that the module is available (plugged in). The module status information that is read here is then available in the form of service output parameters (MOD_INF).

Detailed information about the errors is entered in the DIAG_INF output parameter of data type STRUCT. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; Diagnostic Data, Byte 0 to Byte 8, Structure of Channel-Specific Diagnostic Data*.

---

**Note**

Even if you run a HART module in HART MODE =16#070C, any HART protocol errors/configuration changes that may occur will be masked by the MOD_D3 block, and not signaled as channel errors.

---

Several channel types can occur in one diagnostic data record. If the "further channel types exist" bit is set, MOD_D3 runs through the entire diagnostic evaluation and handles reporting for each further channel type.

HART modules with the firmware update and the new additional channel type 16#66 for measured value calibration are also supported.

The new HART channel type 66 can also occur in the diagnostic data record as a further channel type.

The new diagnostic data ("channel being calibrated" and "channel temporarily not available") are considered as channel errors and output. The detailed texts are output via the system text library and a corresponding associated value.

Channel type 0x66:
1 byte channel diagnostics
Bit0 channel being calibrated
Bit1 channel temporarily not available

## Firmware update

The firmware update for the listed HART modules is started by a diagnostic event "OB83 entering state" (remove module) followed directly by diagnostic event "OB83 exiting state" (insert module). With "OB83 exiting state", byte 2 bit 2 is set in data record 0 (1 = STOP mode).

Once the firmware is completed, there is a repeated diagnostic event "OB83 entering state" (remove module) followed directly by diagnostic event "OB83 exiting state" (insert module). With "OB83 exiting state", byte 2 bit 2 is reset in data record 0 (0 = RUN mode).

In MOD_D3, after an "OB83 exiting state" (module removed) the data record 0 (DS0) is always read extra in OB1 using SFC51 and SZL 00B1 to establish whether bit (1 = STOP mode) is set. If this is the case, this is always recognized as a firmware update and the module continues to be indicated as removed and not available. Only when there is an OB83 (module inserted) with the information in DS0 (0 = RUN mode), is the module indicated as being inserted and available again.

It is assumed that "Module change in run" is always set for the ET 200M head modules so that a firmware update of the HART module always calls an OB83. This means that a firmware update cannot trigger an OB86 diagnostic interrupt.

The "Generate module driver" function enters 16#0001 in the Feature01 parameter for both HART modules. This means that DS0 is read extra when there is an OB 83 exiting state only when Feature01 = 16#0001

The Feature parameters (FEATURE_01 .. FEATURE_10) are intended for future expansions of the MOD_D3 block and for parameter settings for special module situations.

Currently only FEATURE_01 is used as an ID for HART module with a firmware update in RUN.

## Redundancy

The higher-level RACK block monitors the redundancy of DP master systems operating in an H system.

## MODE Setting

You will find more information about this in the "MODE settings (Page 387)" section.

---

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

---

## OMODE Structure

You will find more information about this in the "OMODE settings (Page 396)" section.

## Valid channel display

The available channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find more information about this in "Addressing (Page 399)".

HART modules with read/write access to the process image are configured in the same way as input modules. The set I/O range must always be identical.

Example: SM 332 AO 2x0/4..20mA HART 332-5TB00-0AB0:

| Address input range (HW Config) | Address output range (HW Config) | LADDR (decimal/hex) |
|---|---|---|
| 544 | 544 | 544 / 16#0220 |

## Error handling

The plausibility of input parameters is not checked.
You will find more information about error handling in "Error information of output parameter MSG_STAT" (Page 398).

## Service Information

To analyze faults, the module status information entered during startup is read via the MOD_INF structured output parameter. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; System Status List, Module Status Information*.

Following a diagnostic interrupt, you will also find detailed module diagnostic information in the MODDIAG0 to MODDIAG8 output parameters. You will find more information in the reference manual *System Software for S7-300/400 System and Standard Functions; Diagnostic Data, Byte 0 to Byte 8*.

The CHDIAG00 to CHDIAG15 output parameters contain detailed channel-status information. You will find more information in the reference manual *System Software for S7-300/400 System and Standard Functions; Structure of Channel-Specific Diagnostic Data*.

The system resets this diagnostic information after a diagnostic interrupt has been reported outgoing (no further channel or module errors are queued).

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx outputs.

## Overload Behavior

The MOD_D3 block counts the OB82 calls. The counter is reset in OB1. If more than two OB82 events occur in succession before the cycle control point is reached (OB1), these events are rejected and the message "OB82 DP master failure: x Rack: y Slot: z" is output.

## Time response

Not available

## Message response

MOD_D3 uses ALARM_8P_1 to report module errors. The function also calls ALARM_8P_2 and ALARM_8P_3 which are intended for channel errors. The inputs DELAY1 and DELAY2 are used to delay the output of I/O access error messages. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB 85) before it outputs the message. The message is output only under the condition that no higher-priority error is reported within this delay time. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default.
The message function can be disabled by setting EN_MSG = FALSE.

## Flutter suppression

The "Flutter suppression" function is used to delay the outgoing of a message by a configurable period.

The flutter time is entered at the channel block at the FlutTmIn parameter. The high byte of the DXCHG parameter of the channel blocks contains the flutter time.

Flutter suppression comes into effect when FlutEN = 1 or FlutTmIn > 0 is set at the channel block.

There is only one flutter message per module. The delay times and fault messages are channel-specific. The fault messages are extended by at least the delay time. Flutter exists if the fault messages "Outgoing" and then "Incoming" are present within the delay time.

The last fluttering channel and its set delay time deactivates the flutter message.

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

You will find more information in:

Message texts and associated values of MOD_D3 (Page 171)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.22.2 I/Os of MOD_D3

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible; I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number (x = 1 - 3) | DWORD | 0 |
| FEATURE_xx | Feature parameter (xx = 01 - 04) | WORD | 0 |
| FEATURE_yy | Feature parameter (yy = 05 - 10) | DWORD | 0 |
| **LADDR** | Logic input address of the module | INT | 0 |
| **LADDR1** | Logical output address of the module (if output address is not the same as input address). | INT | 0 |
| MODE_xx | Mode channel xx (xx = 00 - 15) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DIAG_INF | System structure: Diagnostic information | STRUCT | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression | DWORD | 16#00000000 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACKx | Message acknowledgment (x = 1 - 3) | WORD | 0 |
| MSGSTATx | Message error information (x = 1 - 3) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Mode channel (xx = 00 - 15) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_D3 (Page 171)

Maintenance status of MS (Page 403)

## 3.22.3 Message texts and associated values of MOD_D3

### Assignment of message text and message class

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 | 1 | QMODF | Module @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | Module @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | QMODF | Module @1%d@/@2%d@/@3%d@: @5W%t#MOD_D3_TXT@ | S |
| | 4 | | Module @1%d@/@2%d@/@3%d@: Multiple diagnostic interrupt | S |
| | 5 | | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_D3_TXT@ | S |
| | 6 | | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_D3_TXT@ | S |
| | 7 | | Module @1%d@/@2%d@/@3%d@: @5W%t#MOD_D3_TXT@ | F |
| | | | - | |
| EV_ID2 | 1 | - | Module @1%d@/@2%d@/@3%d@: Error channel 00 @4W%t#MOD_D3_TXT@ | S |
| | ... ... | | ... ... | ... ... |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error channel 07 @4W%t#MOD_D3_TXT@ | S |
| EV_ID3 | 1 | - | Module @1%d@/@2%d@/@3%d@: Error channel 08 @4W%t#MOD_D3_TXT@ | S |
| | ... ... | | ... ... | ... ... |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error channel 15 @4W%t#MOD_D3_TXT@ | S |

You will find the message texts and their text numbers in "Text library for MOD_D3" (Page 421).

## Assignment of Associated Values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID1 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number (message 5) of MOD_D3_TXT |
| | 5 | | Text number (message 3) of MOD_D3_TXT |
| | | | |
| EV_ID2 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number (messages 1 - 8) from MOD_D3_TXT |
| | | | |
| EV_ID3 | 1 | MOD_INF.SUBN_ID | DP master system ID (byte) |
| | 2 | MOD_INF.RACK_NO | Rack/station number (byte) |
| | 3 | MOD_INF.SLOT_NO | Slot number (byte) |
| | 4 | | Text number (messages 1 - 8) from MOD_D3_TXT |

## See also

Message Classes (Page 401)

# 3.23 MOD_DRV: Monitoring of drive blocks with diagnostics functions

## 3.23.1 Description of MOD_DRV

### Object name (type + number)

FB 148

- Block interfaces of MOD_DRV (Page 175)

### Area of application

The MOD_DRV block creates a device-specific diagnostics function for blocks of the Drive type.

The following events are processed in the MOD_DRV and the status messages reported in WinCC:

| SwitchType | Message text | Message class in WinCC | Maintenance State | Status (for each interconnectable output) |
|---|---|---|---|---|
| Telegram1/Telegram20 (NAMUR off and on) | Ready to switch on | SA | 00 | 16#80 |
| | Ready for operation | SA | 00 | 16#80 |
| | Error active | S | 07 | 16#00 |
| | Warning active | F | 06 | 16#68 |
| Telegram20 (NAMUR on) | Error in the controller/software | S | 07 | 16#00 |
| | Fault in the power supply network | S | 07 | 16#00 |
| | Overvoltage in the DC Link | S | 07 | 16#00 |
| | Fault in the power supply | S | 07 | 16#00 |
| | Temperature limiter converter | S | 07 | 16#00 |
| | Grounding fault | S | 07 | 16#00 |
| | Motor overload | S | 07 | 16#00 |
| | Error in the communication | S | 07 | 16#00 |
| | External safety shutdown | F | 06 | 16#68 |
| | Error in the speed measurement | S | 07 | 16#00 |
| | Error in the internal communication | S | 07 | 16#00 |
| | Fault in the infeed system | S | 07 | 16#00 |
| | Other fault or error | S | 07 | 16#00 |

## Calling OBs

The cyclic OB 1 and OB 100.

In addition, the block is installed in the cyclic interrupt OB OB3x in which the following signal processing driver block is installed.

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The block is installed when the corresponding SlaveFamily 1 object exists
- Further interconnection takes place in the same way as for the existing MODPAL/X0 blocks of the PCS 7 Basis Library
- The corresponding inputs are read out of HW Config and configured.
- The block is interconnected with the corresponding Drive block through the DXCHG_00, OMODE_00 and O_MS parameters.

## Startup characteristics

Initialization of the ALARM_8P blocks and of the NOTIFY_8P blocks.

## Message response

The block reports errors using ALARM_8P and NOTIFY_8P.

The block generates the following messages in the OBs listed below:

| OB no. | Start event | Message |
|---|---|---|
| 1 | Cyclic processing | Have the ALARM_8P/NOTIFY_8P outputs/messages updated, if necessary |
| 100 | Restart | Initialization of ALARM_8P |

## MODE behavior

The block is informed by the MODE input about which function block or device type is involved. A distinction is made between the device and type or profile respectively.

Only the Drive type is currently implemented. This has the mode XXXX0010.

## 3.23.2 I/Os of MOD_DRV

### I/Os

The factory setting of the block display in CFC is identified in the "I/O" column.

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| Parameter | Description | Type | Default |
|---|---|---|---|
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID1 | Message number 1 | DWORD | 16#00 |
| EV_ID2 | Message number 2 | DWORD | 16#00 |
| **MS** | Maintenance status | DWORD | 16#00 |
| **LINK_ADR** | Link address | BYTE | 0 |
| **PADP_ADR** | Address of the PA field device | BYTE | 16#00 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 16#FF |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 16#FF |

### In-out parameters

| Parameter | Description | Type |
|---|---|---|
| **CPU_DIAG** | System structure: CPU diagnostics | Struct (see OB_BEGIN) |
| **RAC_DIAG** | System structure: RACK diagnostics | Struct (see OB_DIAG1) |

## Output parameters

| Parameter | Description | Type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00 00000 0 |
| DXCHG_00 | Bidirectional data exchange channel | DWORD | 16#00 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 16#00 |
| Error | Error message is active | BOOL | 0 |
| Namur1 | Namur1 message is active | BOOL | 0 |
| Namur2 | Namur2 message is active | BOOL | 0 |
| Namur3 | Namur3 message is active | BOOL | 0 |
| Namur4 | Namur4 message is active | BOOL | 0 |
| Namur5 | Namur5 message is active | BOOL | 0 |
| Namur6 | Namur6 message is active | BOOL | 0 |
| Namur7 | Namur7 message is active | BOOL | 0 |
| Namur8 | Namur8 message is active | BOOL | 0 |
| Namur9 | Namur9 message is active | BOOL | 0 |
| Namur10 | Namur10 message is active | BOOL | 0 |
| Namur11 | Namur11 message is active | BOOL | 0 |
| Namur12 | Namur12 message is active | BOOL | 0 |
| Namur13 | Namur13 message is active | BOOL | 0 |
| Namur14 | Namur14 message is active | BOOL | 0 |
| Namur15 | Namur15 message is active | BOOL | 0 |
| Namur16 | Namur16 message is active | BOOL | 0 |
| MSGSTAT1 | Message error information 1 | WORD | 16#00 |
| MSGSTAT2 | Message error information 2 | WORD | 16#00 |
| MSG_ACK1 | Message acknowledgment 1 | WORD | 16#00 |
| MSG_ACK2 | Message acknowledgment 2 | WORD | 16#00 |
| OMODE_00 | Value status of the PA field device | DWORD | 16#00 |
| O_MS | Maintenance status | DWORD | 16#00 |
| QERR | 1 = program error | BOOL | 1 |
| QRACKF | 1 = PA slave/DP master error | BOOL | 0 |
| Warning | Warning message is active | BOOL | 0 |

### 3.23.3 Message texts and associated values of MOD_DRV

#### Messaging

| Message block | Class | Event |
|---|---|---|
| EV_ID1:SIG1 | AS process control message - fault | Device @1%d@/ @2%d@: Bad, maintenance alarm |
| EV_ID1:SIG2 | AS process control message - fault | Device @1%d@/ @2%d@: Uncertain, maintenance demanded |
| EV_ID1:SIG3 | AS process control message - fault | Device @1%d@/ @2%d@: Good, maintenance required |
| EV_ID1:SIG6 | AS process control message - fault | Device @1%d@/ @2%d@: Failure |
| EV_ID2:SIG2 | Status message - AS | Device @1%d@/ @2%d@: Good, configuration has been changed |
| EV_ID2:SIG3 | Status message - AS | Device @1%d@/ @2%d@: Uncertain, simulation |
| EV_ID2:SIG6 | Status message - AS | Device @1%d@/@2%d@: Bad, local operation/functional check |
| EV_ID2:SIG7 | Status message - AS | Device @1%d@/ @2%d@: Bad, device passivated |

All other messages are not assigned.

1. ALARM_8P
2. NOTIFY_8P

#### Assignment of associated values

| Associated value | Block parameter |
|---|---|
| 1 | Subnet |
| 2 | Address |

## 3.24 MOD_HA: Monitoring device-specific diagnostics of HART field devices

### 3.24.1 Description of MOD_HA

#### Object name (type + number)

FB 97

- MOD_HA block I/Os (Page 185)

#### Area of application

The MOD_HA module reports diagnostic events of an HART field device that is connected to a channel of an ET 200M HART module or ET 200iSP HART module. HART modules of ET 200iS are not supported. H systems support only the modules installed in switched racks.

#### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The block is installed downstream of the diagnostic block that is responsible for the HART module.
- LADDR (logical base address of the HART module) is configured.
- The geographic addresses SUBN1_ID, SUBN2_ID, RACK_NO, SLOT_NO, and CHAN_NO (channel number of the HART module to which the HART field device is connected) are configured.
- The CPU_DIAG structures of the OB_BEGIN block are interconnected
- The EN input is interconnected with the output of an AND block.

  The block inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block, and EN_Mxx (xx = module number) of the RACK block and MODE with OMODE_xx of the diagnostic block.
- The DXCHG output parameter is interconnected with the following channel block at the DataXchg parameter.
- The O_MS output parameter is interconnected with the following channel block at the MS parameter.

## Function and method of operation

Block MOD_HA analyzes events relating to a HART field device acyclically. ALARM_8P is used to report these events. The message function can be disabled.

The block is enabled to run by the higher-level RACK block. By default, MOD_HA requires synchronous diagnostic data in OB 82 (additional alarm information with SFB 54 call by OB_BEGIN). With an ET 200iSP HART module, the channel type 16#65 is generated during diagnostics. Here, 2 bytes of diagnostics information are defined per channel of the module, and the block signals the statuses of the 2nd bit of the 1st and 2nd byte for the relevant HART field device.

**Structure of byte 1 (ET 200iSP HART):**

| Bit | Meaning |
|---|---|
| 0 | Parameter assignment error (HART module) |
| 1 | HART communications error (HART module) |
| 2 | Readback error (HART module) |
| 3 | Short circuit (HART module) |
| 4 | Wire break (HART module) |
| 5 | No load voltage (HART module) |
| 6 | Overflow (HART module) |
| 7 | Underflow (HART module) |

Structure of byte 2 (ET 200iSP HART):

| Bit | Meaning |
|---|---|
| 0 | Primary variable outside limits (field device) |
| 1 | Secondary variable outside limits (field device) |
| 2 | Analog output saturated (field device) |
| 3 | Analog output current specified (field device) |
| 4 | More statuses available (field device) |
| 5 | Reserved for maintenance alarm (field device) |
| 6 | Reassignment the field device parameters |
| 7 | Malfunction of the field device |

For an ET 200M with two-channel HART modules, channel type 16#61 or 16#63 is generated during diagnostics. Bit 5 in byte 8 for channel 0 and byte 9 for channel 1 in the additional alarm information means "HART channel error". If bit 5 = TRUE, the additional diagnostic data is read with SFB 52 (RDREC) as follows:

- with data record 128 for channel 0
- with data record 129 for channel 1

Diagnostic data records 128 (for channel 0) and 129 (for channel 1) have the same structure. and return detailed HART diagnostic information on the previous transfer. The table below shows the individual error messages/warnings.

| Byte/Bit No. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0: general | 1= Mod. comm. | No. of (triggering) client, when module comm. no. =0 | | | Polling address (of HART transducer), always 0 for monodrop | | | |
| 1: fault groups = group error | Channel fault (L+, DrBr) | HART channel fault | HART slave communi-cation | HART command error | device status <> 0 (e.g., configure-tion changed) | more status | Com-mand rejected | 0 = Not used |
| then → bytes | - | 2 | 8 | 8 | 9 | - | - | - |
| 2: HART cf = "communi-cation faults" Field device for module | HART access not possible | parity error in response | overrun error in response | framing error in response | wrong checksum in response | Wrong char timing | too many chars in respon-se | wrong telegram timing |
| 3 to 6: time stamp | Broadcast system time: Milliseconds (10s and 100s), seconds, minutes, and hours in two-digit BCD code. If the timestamp function is not used: Content = 0 | | | | | | | |
| 7: HART/module | last HART or module command | | | | | | | |
| 8: HART ce | 1<br>0 | "Communication error bits" of "slave", (first status byte)<br>"Command response" list - no errors, but warnings | | | | | | |
| 9: HART ds | Device status bits (second status byte) | | | | | | | |

Two HART status bytes are reserved in the HART protocol to display errors and warnings. These are entered in diagnostic data records 128 and 129 unchanged. The meaning of the HART status bytes is defined in the HART Standard.

- First HART status byte (meaning depends on bit 7):
  - Bit 7 = 1: Communication error during the transmission of a HART command to the field device
  - Bit 7 = 0: Only warnings that the field device sends in response to a command

<table>
<tr><th>Bit 7/Bit No.</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>Either<br>Bit 7 = 1:<br>HART<br>"communication error" from module to field device</td><td>1</td><td>parity error in command</td><td>overrun error in command</td><td>framing error in command</td><td>wrong checksum in command</td><td>Reserved = 0</td><td>too many characters in command<br>(rx buffer overflow)</td><td>(un-defined)</td></tr>
<tr><td>or<br>Bit 7 = 0:<br>HART<br>"response to a command"</td><td>0</td><td colspan="7">The messages in bits 0 - 6 are coded as integers:<br>0: No command-specific error<br>1: Undefined<br>2: Invalid section<br>3: Transferred parameter too large<br>4: Transferred parameter too small<br>5: Too few data bytes received<br>6: Device-specific command error (rarely used)<br>7: In write-protected mode<br>8-15: Various meanings (see code commands)<br>16: Restricted access<br>28: Various meanings (see code commands)<br>32: Device is busy<br>64: Command not implemented</td></tr>
</table>

| Code | Commands | Alternative meanings |
|---|---|---|
| 8 *) | 1,2,3,33,60,61,62,<br>110,34,55,64,48 | "Update" error<br>set to nearest possible value,<br>"Update" being executed |
| 9 | 35,65,36,37,43,52,45,46,67,68 | Lower range limit too high,<br>"applied process" too high,<br>not in correct current mode<br>(fixed at 4 mA or 20 mA) |
| 10 | 6,35,65,36,37,43,52 | Multidrop is not supported<br>Lower range limit too low,<br>"applied process" too low, |
| 11 | 35,65,40,45,46,66,67,68,53 | Upper range limit too high,<br>In multidrop mode,<br>bad transmitter variable code |
| 12 | 35,65,53,66,67,68 | Upper range limit too low,<br>bad units code |
| 13 | 35,65,69 | Both range limits outside the limit value,<br>bad transfer function code |
| 14 *) | 35,36,65,37 | Span too limited,<br>"pushed" upper range limit outside the limit |
| 15 | 65,66,67,68,69 | Faulty code for the number of the analog output |
| 28 | 65 | bad code for the range unit<br>("range units code") |

- Second HART status byte: Device status of the HART field device in the event of a communication error
  (otherwise, byte = 0)

| Bit No. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| HART device status:<br>"field device status" | Malfunction of the field device | Reassign-ment of parameters:<br>"configure-tion changed (CC)" | Cold restart | Further status available<br>"more status" | Analog output current specified<br>("fixed)" | Analog output saturated | Non-primary variable outside limits | Primary variable outside limits |

Process control messages are generated when "communication errors" and HART field device errors (byte 9 <> 0) occur. Operating messages with acknowledgment are generated if bit 7 = 0 (byte 8) and the remaining bits <> 0. The last read data record 128 or 129 (depending on the channel number) is written to the output structure DIAG_H.

Bytes 8 and 9 are evaluated and event messages generated in OB1.
You will find additional information in "Message texts and associated values of MOD_HA (Page 187)".

The MODE input is interconnected with the corresponding OMODE_xx output of the diagnostic block. The module channel configurations set in HW Config are reported at these locations. MODE (Page 387) is written to the low word of theOMODE (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If valid, OMODE = 16#80xxxxxx. The diagnostic block contains the events that lead to a value status "invalid value due to higher-priority error" (OMODE = 16#40xxxxxx), or to channel error (OMODE = 16#00xxxxxx).

## Redundancy

The higher-level RACK block evaluates the redundancy of DP master systems operating in an H system. Redundant HART field devices are not supported.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

A restart (OB 100) is reported via the LSB in byte 2 of the OMODE (Page 396) output.

ALARM_8P will be initialized.

## Overload behavior

The MOD_HA block counts the OB 82 calls. The counter is reset in OB1. A diagnostic message will not be generated if more than five OB 82 events occur before the cycle control point is reached (OB 1). A "multiple diagnostic interrupt" message will not be generated, because the diagnostic block performs this action.

## Time response

Not available

## Message response

MOD_HA reports diagnostic information from a HART field device by means of ALARM_8P or NOTIFY_8P.

The message function can be disabled by setting EN_MSG = FALSE.

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_HA (Page 187)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.24.2 I/Os of MOD_HA

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Type | De-fault |
|---|---|---|---|
| **CHAN_NO** | Channel number | BYTE | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| **MODE** | Channel operating mode | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#0 0000 000 |
| DIAG_H | Diagnostic information of HART communication channel | STRUCT | |
| DXCHG | Bidirectional data exchange channel<br>Bit 0 = Release for maintenance<br>Bits 1-31 = Reserve | DWORD | 0 |
| **EXT_STAT** | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression | DWORD | 16#0 0000 000 |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSGSTAT | Message error information | WORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| O_MS | Maintenance status | DWORD | 0 |
| **OMODE** | Channel operating mode | DWORD | 0 |

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QREC_ERR | 1 = Read diagnostic data error | BOOL | 0 |
| QREC_VAL | 1 = Read diagnostic data | BOOL | 0 |
| STATUS | Read diagnostics status | DWORD | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | CPU diagnostics | STRUCT | |
| DXCHG_IN | Bidirectional data exchange channel | DWORD | 0 |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_HA (Page 187)

Maintenance status of MS (Page 403)

### 3.24.3 Message texts and associated values of MOD_HA

#### Assignment of message text and message class

| Message block | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID (ALARM_8P) | 1 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Communication errors | S |
| | 2 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Errors | S |
| | 3 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Secondary var. outside range | F |
| | 4 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Primary var. outside range | F |
| | 5 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Analog value specified | S |
| | 6 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Analog value saturated | S |
| | 7 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Maintenance alarm | S |
| | 8 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Further status available | F |
| | | | |
| EV_ID1 (NOTIFY_8P) | 1 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Reassignment of parameters | SA |
| | 2 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Cold restart | SA |
| | 3 | | No message |
| | 4 | | No message |
| | 5 | | No message |
| | 6 | | No message |
| | 7 | | No message |
| | 8 | | No message |

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |
| | 4 | CHAN_NO | Channel error text number |
| | | | |
| EV_ID1 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number |
| | 3 | SLOT_NO | |
| | 4 | CHAN_NO | |

If SUBN1_ID = 16#FF, the associated value is substituted with SUBN2_ID.

## See also

Message Classes (Page 401)

# 3.25 MOD_MS: Monitoring up to 16 channels on ET200S/X motor starter modules with diagnostic functions

## 3.25.1 Description of MOD_MS

### Object name (type + number)

FB 96

- MOD_MS block I/Os (Page 194)

### Area of application

The MOD_MS block the up to 16-channel motor starter modules with diagnostic capability (ET 200S). H systems support only the modules installed in switched racks.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The MOD_MS block is installed in its runtime group downstream of the RACK block runtime group in the above-mentioned OBs.
- The MODE_xx, SUBN1_ID, SUBN2_ID, and SUBN_TYP inputs are configured.
- The logical addresses LADDR and LADDR1 are configured.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of MOD_MS.
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function and method of operation

Block MOD_MS analyzes all events that affect a module and its channels acyclically. It generates a channel-specific MODE (Page 387) and value status for the signal processing blocks. ALARM_8P is used to report these events. The message function can be disabled.

The block is enabled to run by the higher-level RACK block. The event to be evaluated is stored in the CPU_DIAG start and diagnostic information of the OB_BEGIN block. There is a MODE_xx input for each signal channel of the module. The module channel configuration data created in HW Config is reported here. The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current channel value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx.

The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Events that are evaluated by the RACK block:
  - Rack failure (OB 86) (output parameter QRACKF = TRUE)
  - Program execution error (OB 85) (output parameter QRACKF = TRUE)
- Events that are evaluated by the MOD block:
  - Program execution error (OB 85) (output parameter QPERAF = TRUE)
  - Module removed (OB 83) (output parameter QMODF = TRUE)
  - Diagnostic interrupt (OB 82)

The following events in OB 82 will lead to a module error, and are indicated with 16#40xxxxxx ("higher-priority error") in OMODE. At the same time, output parameter QMODF = TRUE:

- Configuration/parameter assignment error
- Overload
- Short circuit
- Error
- Actuator OFF
- Wire break
- Safety-related shutdown
- High limit exceeded
- Low limit undershot
- Missing supply voltage
- Switching element overload
- External error

ALARM_8P is used to report "module removed", "I/O access error", and the above "OB 82 error" events to WinCC.

The system verifies during startup that the module is available (plugged in). The module status information that is read here is then available in the form of service output parameters (MOD_INF).

Detailed information about the errors is entered in the DIAG_INF output parameter of data type STRUCT.
You will find additional information in the "Service Information" section, and in the *System Software for S7-300/400 System and Standard Functions; Diagnostic Data, Byte 0 to Byte 8, Structure of Channel-Specific Diagnostic Data* reference manual.

## Redundancy

The block supports segment redundancy of CPU 417H for distributed I/Os. To use this function, you must configure the SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs with the numbers of the redundant segments. If there is no segment redundancy, the remaining input must be set to the (default) value 16#FF.

## MODE setting

You will find more information about this in the "MODE settings (Page 387)" section.

---

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

---

## OMODE Structure

You will find additional information in the "OMODE (Page 396)" section.

## Valid channel display

The existing channels on a module are displayed in the CH_EXIST output by setting a bit in the DWORD, starting at bit 0, for every existing channel. If the bit assigned to a channel = 0, the channel is not available.

Output CH_OK displays the valid channels on a module by setting a bit to TRUE for every valid channel, where bit 0 is assigned to channel 0, etc. If the bit assigned to a channel is 0, the channel is faulty. If a module error occurs, all channels are disrupted.

## Addressing

You will find additional information in the "Addressing (Page 399)" section.

## Error handling

The plausibility of input parameters is not checked.
You will find more information about this in "Error information of output parameter MSG_STAT (Page 398)".

## Service Information

To analyze faults, the module status information entered during startup is read via the MOD_INF structured output parameter. You will find more information about this in the reference manual *System Software for S7-300/400 System and Standard Functions; System Status List, Module Status Information*.

Following a diagnostic interrupt, you will also find detailed module diagnostic information in the MODDIAG0 to MODDIAG8 output parameters. You will find more information in the reference manual *System Software for S7-300/400 System and Standard Functions; Diagnostic Data, Byte 0 to Byte 10*.

The CHDIAG00 to CHDIAG15 output parameters contain detailed channel-status information. You will find more information in the reference manual *System Software for S7-300/400 System and Standard Functions; Structure of Channel-Specific Diagnostic Data*.

Of the motor starter module channels, only channel 0 is assigned the diagnostic function. The error code is stored in CHDIAG00 to CHDIAG03. You will find additional information about this in the *ET 200S, Motor Starter Safety Technology SIGUARD; Diagnostics and Monitoring via the User Program* reference manual.

The system resets this diagnostic information after a diagnostic interrupt has been reported outgoing (no further channel or module errors are queued).

## Startup characteristics

After a restart/initial startup, the system verifies that the module is available under its logical base address. A restart (OB 100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

MOD_MS reports module and motor-starter errors by means of ALARM_8P_1 and ALARM_8P_2. The DELAY1 and DELAY2 inputs are used to delay the I/O access error message. DELAY1 allows you to enter a time in seconds for which the block will wait for a higher-priority error (rack failure or removal/insertion) following a program execution error (OB 85) before it outputs the message. DELAY2 determines the number of seconds the block waits after the higher-priority error has been reported outgoing until it outputs the queued I/O access error as well. Both values are set to 2 seconds by default.

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_MS (Page 196)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.25.2 I/Os of MOD_MS

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number x | DWORD | 0 |
| **LADDR** | Logical address of the input channels | INT | 0 |
| **LADDR1** | Logical address of the output channels | INT | 0 |
| MODE_xx | Mode channel xx (xx = 00 - 31) | WORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DIAG_INF | System structure: Diagnostic information | STRUCT | |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 31)<br>Bit 0 = Release for maintenance<br>Bit 1-31 = Reserve | DWORD | 0 |
| **EXT_STAT** | Release for maintenance - extended status | DWORD | 0 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACKx | Message acknowledgment x | WORD | 0 |
| MSGSTATx | Message error information x | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| OMODE_xx | Mode channel xx (xx = 00 - 31) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = failure of the rack | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_MS (Page 196)

Maintenance status of MS (Page 403)

### 3.25.3 Message texts and associated values of MOD_MS

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 | 1 | QMODF | Module @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | QPERAF | Module @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_MS_TXT@ | S |
| | 4 | - | Module @1%d@/@2%d@/@3%d@: Multiple diagnostic interrupt | S |
| | 5 | - | Module @1%d@/@2%d@/@3%d@: @4W%t#MOD_MS_TXT@ | S |
| | 6 | - | Module @1%d@/@2%d@/@3%d@: Short-circuit | S |
| | 7 | - | Module @1%d@/@2%d@/@3%d@: Overload | S |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Error | S |
| EV_ID2 | 1 | | Module @1%d@/@2%d@/@3%d@: High limit exceeded | S |
| | 2 | | Module @1%d@/@2%d@/@3%d@: Low limit undershot | S |
| | 3 | | Module @1%d@/@2%d@/@3%d@: Parameter assignment error | S |
| | 4 | - | Module @1%d@/@2%d@/@3%d@: Actuator OFF | S |
| | 5 | - | Module @1%d@/@2%d@/@3%d@: Safety-related shutdown | S |
| | 6 | - | Module @1%d@/@2%d@/@3%d@: External error | S |
| | 7 | - | Module @1%d@/@2%d@/@3%d@: Switching element overload | S |
| | 8 | - | Module @1%d@/@2%d@/@3%d@: Missing supply voltage | S |

You will find the message texts and their text numbers in "Text library for MOD_MS (Page 424)".

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID1 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |
| | 4 | - | Text number (message 5) of MOD_MS_TXT |
| | | | |
| EV_ID2 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |

## See also

Message Classes (Page 401)

# 3.26 MOD_SWT: Monitoring of Switch blocks with diagnostics functions

## 3.26.1 Description of MOD_SWT

### Object name (type + number)

FB 149

- MOD_SWT block I/Os (Page 200)

### Area of application

The MOD_SWT block creates the device-specific diagnostics function for blocks of the Switch type.

The following events are processed in the MOD_SWT and the status messages reported in WinCC:

| SwitchType | Message | Message Class WinCC | MS | Status (for each interconnectable output) |
|---|---|---|---|---|
| FbSwtMMS | Run reverse | SA | 00 | 16#80 |
| | Off | SA | 00 | 16#80 |
| | Run forward | SA | 00 | 16#80 |
| | Overload warning | F | 06 | 16#68 |
| | Fault | S | 07 | 16#00 |
| | Warning | F | 06 | 16#68 |

### Calling OBs

The cyclic OB 1 and OB 100.

In addition, the block is installed in the cyclic interrupt OB OB3x in which the following signal processing driver block is installed.

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The block is installed when the corresponding SlaveFamily 2 object exists
- Further interconnection takes place in the same way as for the existing MODPAL/X0 blocks of the PCS 7 Basis Library
- The corresponding inputs are read out of HW Config and configured.
- The block is interconnected with the corresponding Switch block through the DXCHG_00, OMODE_00 and O_MS parameters.

## Startup characteristics

Initialization of the ALARM_8P blocks and of the NOTIFY_8P blocks.

## Message response

The block reports errors using ALARM_8P and NOTIFY_8P.

The block generates the following messages in the OBs listed below:

| OB no. | Start event | Message |
|---|---|---|
| 1 | Cyclic processing | Have the ALARM_8P/NOTIFY_8P outputs/messages updated, if necessary |
| 100 | Restart | Initialization of ALARM_8P |

## MODE behavior

The block is informed by the "MODE" input about which function block or device type is involved.

## 3.26.2 I/Os of MOD_SWT

### I/Os

The factory setting of the block display in CFC is identified in the "I/O" column.

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID1 | Message number 1 | DWORD | 16#00 |
| EV_ID2 | Message number 2 | DWORD | 16#00 |
| **MS** | Maintenance status | DWORD | 16#00 |
| **LINK_ADR** | Link address | BYTE | 0 |
| **PADP_ADR** | Address of the PA field device | BYTE | 16#00 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 16#FF |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 16#FF |

### In-out parameters

| I/O (parameter) | Meaning | Type |
|---|---|---|
| **CPU_DIAG** | System structure: CPU diagnostics | Struct (see OB_BEGIN) |
| **RAC_DIAG** | System structure: RACK diagnostics | Struct (see OB_DIAG1) |

## Output parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| DXCHG_00 | Bidirectional data exchange channel | DWORD | 16#00 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 16#00 |
| Error | Error message is active | BOOL | 0 |
| MSGSTAT1 | Message error information 1 | WORD | 16#00 |
| MSGSTAT2 | Message error information 2 | WORD | 16#00 |
| MSG_ACK1 | Message acknowledgment 1 | WORD | 16#00 |
| MSG_ACK2 | Message acknowledgment 2 | WORD | 16#00 |
| OMODE_00 | Value status of the PA field device | DWORD | 16#00 |
| O_MS | Maintenance status | DWORD | 16#00 |
| Overload | Overload message is active | BOOL | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QRACKF | 1 = PA slave/DP master error | BOOL | 0 |
| Warning | Warning message is active | BOOL | 0 |

### 3.26.3 Message texts and associated values of MOD_SWT

#### Messaging

| Message block | Class | Event |
|---|---|---|
| EV_ID1:SIG1 | AS process control message - fault | Device @1%d@/ @2%d@: Bad, maintenance alarm |
| EV_ID1:SIG2 | AS process control message - fault | Device @1%d@/ @2%d@: Uncertain, maintenance demanded |
| EV_ID1:SIG3 | AS process control message - fault | Device @1%d@/ @2%d@: Good, maintenance required |
| EV_ID1:SIG6 | AS process control message - fault | Device @1%d@/ @2%d@: Failure |
| EV_ID2:SIG2 | Status message - AS | Device @1%d@/ @2%d@: Good, configuration has been changed |
| EV_ID2:SIG3 | Status message - AS | Device @1%d@/ @2%d@: Uncertain, simulation |
| EV_ID2:SIG6 | Status message - AS | Device @1%d@/@2%d@: Bad, local operation/functional check |
| EV_ID2:SIG7 | Status message - AS | Device @1%d@/ @2%d@: Bad, device passivated |

All other messages are not assigned.

1. ALARM_8P
2. NOTIFY_8P

#### Assignment of associated values

| Associated value | Block parameter |
|---|---|
| 1 | Subnet |
| 2 | Address |

# 3.27 MOD_PAL0: Diagnosing a DPV0 PA slave (via DP/PA coupler downstream of a DP/PA link DPV1)

## 3.27.1 Description of MOD_PAL0

### Object name (type + number)

FB 99

- MOD_PAL0 block I/Os (Page 206)

### Area of application

Block MOD_PAL0 reports the maintenance status of a PA field device that is used as a DPV0 slave downstream of a DP/PA link DPV1. The PA field devices must conform to the PROFIBUS V3.0 profile.

### Calling OBs

The cyclic OB and OB 100.

In addition, the block is installed in the cyclic interrupt OB OB3x in which the following signal processing driver block is installed.

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The block is integrated in the run sequence upstream of block PA_x
- The SUBN1_ID, SUBN2_ID, RACK_NO, SLOT_NO, PADP_ADR, PROF_V30 inputs are configured.
- The block inputs are interconnected with the following outputs:
  – Output PA_DIAG of block PADP_L10
  – OMODEx outputs of block PADP_L10
  – QMODF and QPERAF outputs of block PADP_L10
  – with output structure RAC_DIAG of block OB_DIAG1
  – input QC_x with PA field device icon

- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

**Note**

The CFC function "**Generate module drivers**" can only be used if the PA field device belongs to slave family 12.

## Function and method of operation

Block MOD_PAL0 analyzes all events that affect the status of a PA field-device slot cyclically. With modular PA field devices, the statuses of the slots are combined to form one status. The acyclic diagnostic events of a PA field device are acquired by the PADP_L10 block. It then stores them in the PA_DIAG parameter. The PA field-device status and the diagnostic information are evaluated, and entered in the MS parameter.

For more information on this, refer to the section:
"PA field device status and diagnostic information (Page 413)".

The statuses are generated with ALARM_8P for messages requiring acknowledgment, and with NOTIFY_8P for those not requiring acknowledgment. The message function can be disabled.

Input PROF_V30 must be set to zero if the PA field device used does not conform to profile 3.0 (this is done by the "Generate module drivers" CFC function).

For a diagnostic event, the block reports "Device xx: uncertain diagnosis".

The failure of a PA field device is identified in the upstream block OB_DIAG1, and is reported via the RAC_DIAG structure. A message "Device xx: Failure" is also generated.

## Redundancy

The higher-level block evaluates the redundancy of DP master systems operating in an H system.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

Initialization of ALARM_8P and NOTIFY_8P.

## Time response

Not available

## Message response

The block reports by means of ALARM_8P and NOTIFY_8P.

The block generates the following messages in the OBs listed below:

| OB no. | Start Event | Message |
|---|---|---|
| x | Cyclic processing | Have the ALARM_8P/NOTIFY_8P outputs/messages updated, if necessary |
| 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values for MOD_PAL0 (Page 208)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.27.2 I/Os of MOD_PAL0

### I/Os

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| **EN_DIAG** | 1 = Queued diagnostic event | BOOL | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number x | DWORD | 0 |
| MODE_xx | Value status PA field device (xx = channel 16 to 31) | DWORD | 0 |
| **MODF** | 1 = PA slave error | BOOL | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| NUM_CHN | Number of channels of the PA device | INT | 0 |
| **PA_DIAG** | PA field device diagnostic information | DWORD | 0 |
| **PADP_ADR** | Address of the PA field device | BYTE | 0 |
| **PERAF** | 1 = I/O access error | BOOL | 0 |
| **PROF_V30** | 1 = PA slave profile V3.0 | BOOL | 0 |
| QC_xx | Status of PA field device (xx = channel 16 to 31) | BYTE | 0 |
| **RACK_NO** | Number of the DP link | BYTE | 0 |
| **SLOT_NO** | Slot number of the PA field device in the DP link | BYTE | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

## Output parameters

| I/O | Meaning | Type | Default |
|---|---|---|---|
| B_QC | Bit-granular Σ status (channel 0 to 31)<br>of the PA field device | STRUCT | |
| CH_ACTIVE | Channel active | DWORD | 16#000<br>00000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression | DWORD | 16#000<br>00000 |
| MSG_ACKx | Message acknowledgment x | WORD | 0 |
| MSGSTATx | Message error information x | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Operating mode slot (xx = Slot 16 to 31) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = PA slave error | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = PA slave/DP master error | BOOL | 0 |

## In-out parameters

| I/O | Meaning | Type | Default |
|---|---|---|---|
| RAC_DIAG | System structure: RACK diagnostics | STRUCT | 0 |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values for MOD_PAL0 (Page 208)

Maintenance status of MS (Page 403)

PA field device status and diagnostic information (Page 413)

### 3.27.3 Message texts and associated values of MOD_PAL0

#### Assignment of message text and message class

| Message block | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 (ALARM_8P) | 1 | Device @1%d@/ @2%d@/@3%d@: Bad, maintenance alarm | S |
| | 2 | Device @1%d@/ @2%d@/@3%d@: Uncertain, maintenance request | F |
| | 3 | Device @1%d@/ @2%d@/@3%d@: Good, maintenance required | M |
| | 4 | Device @1%d@/ @2%d@/@3%d@: Access error | S |
| | 5 | Device @1%d@/ @2%d@/@3%d@: Uncertain diagnostics | S |
| | 6 | Device @1%d@/ @2%d@/@3%d@: Failure | S |
| | 7 | | No message |
| | 8 | | No message |
| | | | |
| EV_ID2 (NOTIFY_8P) | 1 | Device @1%d@/ @2%d@/@3%d@: Good, changes to fail-safe position | SA |
| | 2 | Device @1%d@/ @2%d@/@3%d@: Good, configuration has been changed | SA |
| | 3 | Device @1%d@/ @2%d@/@3%d@: Uncertain, simulation | SA |
| | 4 | Device @1%d@/ @2%d@/@3%d@: Uncertain, due to process, no maintenance | SA |
| | 5 | Device @1%d@/ @2%d@/@3%d@: Bad, process-related, no maintenance | SA |
| | 6 | Device @1%d@/ @2%d@/@3%d@: Bad, local operation/functional check | SA |
| | 7 | Device @1%d@/ @2%d@/@3%d@: Bad, device passivated | SA |
| | 8 | Device @1%d@/ @2%d@/@3%d@: @4W%t#MOD_PAL0_TXT@ | SA |

## Assignment of associated values

| Message block | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID1 (ALARM_8P) | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADPADR | PA field device address (byte) |
| | | | |
| EV_ID2 (NOTIFY_8P) | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADPADR | PA field device address (byte) |
| | 4 | | Text number MOD_PAL0 |

If the PA field device is connected downstream from an inactive DP/PA-Link V1and SUBN1_ID = 16#FF, the associated variable is substituted by SUBN2_ID.

You will find the message texts and their text numbers in "Text library for MOD_PAL0 (Page 416)".

## See also

Message Classes (Page 401)

# 3.28 MOD_PAX0: Diagnosing a DPV0 PA slave (via DP/PA coupler with connection to a DP master system)

## 3.28.1 Description of MOD_PAX0

### Object name (type + number)

FB 112

- MOD_PAX0 block I/Os (Page 213)

### Area of application

Block MOD_PAX0 reports the maintenance status of a PA field device that is used as a DPV0 slave in a DP master system. The PA field devices must conform to the PROFIBUS V3.0 profile.

### Calling OBs

The cyclic OB and OB 100.

In addition, the block is installed in the cyclic interrupt OB OB3x in which the following signal processing driver block is installed.

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The block is integrated in the run sequence upstream of the PA_x block.
- The inputs SUBN1_ID, SUBN2_ID, PADP_ADR and PROF_V30 have parameters assigned.
- The inputs are interconnected with the following outputs:
  - Output PA_DIAG of block PADP_L10
  - OMODEx outputs of block PADP_L10
  - QMODF and QPERAF outputs of block PADP_L10
  - with output structure RAC_DIAG of block OB_DIAG1
- Input QC_x is interconnected with the PA field device status icon.
- Output OMODExx is interconnected with the MODE input of the PA_x block.
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

**Note**

The CFC function "**Generate module drivers**" can only be used if the PA field device belongs to slave family 12.

## Function and method of operation

Block MOD_PAX0 analyzes all events relating to the status of a PA field device slot cyclically. With modular PA field devices, the statuses of the slots are combined to form one status. The acyclic diagnostic events of a PA field device are acquired by the PADP_L10 block. It then stores them in the PA_DIAG parameter. The PA field-device status and the diagnostic information are evaluated, and entered in the MS parameter.

You will find more information in the "PA field device status and diagnostic information (Page 413)" section.

Input PROF_V30 must be set to zero if the PA field device used does not conform to profile 3.0 (this is done by the "Generate module drivers" CFC function).

In the event of a diagnostic event, the block reports "PA field device diagnostics".

There is an input (MODE_xx (Page 387)) for each slot (module) on the PA field device that is used to read in configuration settings made for the PA field device slots (module) in HW Config.

The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current slot value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx. The following events lead to the value status "Invalid value due to higher-priority error" (OMODE_xx = 16#40xxxxxx):

## MODE setting for PA profiles

You will find more information in "PA_MODE settings (Page 397)".

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

## OMODE Structure

You will find additional information in the "OMODE (Page 396)" section.

## Redundancy

The higher-level block evaluates the redundancy of DP master systems operating in an H system.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

Initialization of ALARM_8P and NOTIFY_8P

## Time response

Not available

## Message response

The block uses ALARM_8P and NOTIFY_8P

The block generates the following messages in the OBs listed below:

| OB no. | Start Event | Message |
|---|---|---|
| x | Cyclic processing | Repeat the update of ALARM_8P outputs/messages, if necessary |
| 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values for MOD_PAX0 (Page 215)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.28.2 I/Os of MOD_PAX0

### I/Os

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number x | DWORD | 0 |
| MODE_xx | Value status PA field device (xx = channel 16 to 31) | DWORD | 0 |
| **MODF** | 1 = PA slave error | BOOL | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| NUM_CHN | Number of channels of the PA device | INT | 0 |
| **PA_DIAG** | PA field device diagnostic information | DWORD | 0 |
| **PADP_ADR** | Address of the PA field device | BYTE | 0 |
| **PERAF** | 1 = I/O access error | BOOL | 0 |
| **PROF_V30** | 1 = PA slave profile V3.0 | BOOL | 0 |
| QC_xx | Status of PA field device (xx = channel 16 to 31) | BYTE | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| B_QC | Bit-granular Σ status (channel 0 to 31) of the PA field device | STRUCT | |
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression | DWORD | 16#000000000 |
| MSG_ACKx | Message acknowledgment x | WORD | 0 |
| MSGSTATx | Message error information x | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Operating mode slot (xx = Slot 16 to 31) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = PA slave error | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = PA slave/DP master error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| RAC_DIAG | System structure: RACK diagnostics | STRUCT | 0 |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values for MOD_PAX0 (Page 215)

Maintenance status of MS (Page 403)

PA field device status and diagnostic information (Page 413)

### 3.28.3 Message texts and associated values of MOD_PAX0

#### Assignment of message text and message class

| Message block | Message no. | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 (ALARM_8P) | 1 | - | Device @1%d@/ @2%d@:<br>Bad, maintenance alarm | S |
| | 2 | - | Device @1%d@/ @2%d@:<br>Uncertain, maintenance request | F |
| | 3 | - | Device @1%d@/ @2%d@:<br>Good, maintenance necessary | M |
| | 4 | - | Device @1%d@/ @2%d@:<br>Access error | S |
| | 5 | - | Device @1%d@/ @2%d@:<br>Undefined diagnostics | S |
| | 6 | - | Device @1%d@/ @2%d@:<br>Failure | S |
| | 7 | - | | No message |
| | 8 | - | | No message |
| | | | | |
| EV_ID2 (NOTIFY_8P) | 1 | - | Device @1%d@/ @2%d@:<br>Good, changes to fail-safe position | SA |
| | 2 | - | Device @1%d@/ @2%d@:<br>Good, configuration has been changed | SA |
| | 3 | - | Device @1%d@/ @2%d@:<br>Uncertain, simulation | SA |
| | 4 | - | Device @1%d@/ @2%d@:Uncertain, due to process, no maintenance | SA |
| | 5 | - | Device @1%d@/ @2%d@:Bad, process-related, no maintenance | SA |
| | 6 | - | Device @1%d@/ @2%d@:Bad, local operation/functional check | SA |
| | 7 | - | Device @1%d@/ @2%d@:<br>Bad, device passivated | SA |
| | 8 | - | Device @1%d@/<br>@2%d@:@3W%t#MOD_PAX0_TXT@ | SA |

## Assignment of associated values

| Message block | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID1 (ALARM_8P) | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | PADPADR | PA field device address (byte) |
| | | | |
| EV_ID2 (NOTIFY_8P) | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | PADPADR | PA field device address (byte) |
| | 3 | | Text number MOD_PAXL0 |

If SUBN1_ID = 16#FF, the associated value is substituted with SUBN2_ID.

You will find the message texts and their text numbers in the

"Text library for MOD_PAL0 (Page 416)" section.

## See also

Message Classes (Page 401)

# 3.29 OB_BEGIN: CPU Diagnostics and AS Connection Diagnostics

## 3.29.1 Description of OB_BEGIN

### Object name (type + number)

FB100

- OB_BEGIN block I/Os (Page 222)

### Area of application

Block OB_BEGIN is used for CPU diagnostics of the automation system (AS). By installing the block in CFC, the system creates all acyclic run sequences (OBs) in which the driver blocks of PCS 7 Advanced Process Library are executed.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic processing |
| OB 55 | Status interrupt (only if a DP/PA slave is required) |
| OB 56 | Update interrupt (only if a DP/PA slave is required) |
| OB 57 | Vendor-specific alarm (only if a DP/PA slave is required) |
| OB 70 | I/O redundancy error |
| OB 72 | CPU redundancy error |
| OB 80 | Timeout error |
| OB 81 | Power supply error |
| OB 82 | Diagnostic interrupt |
| OB 83 | Remove/insert interrupt |
| OB 84 | CPU hardware error (only for CPU with this function) |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 88 | Stop avoidance |
| OB 100 | Restart |
| OB 121 | Programming error |
| OB 122 | I/O access error |

### Use in CFC

With the "**Generate module drivers**" CFC function, the OB_BEGIN block is automatically installed in the OBs listed above.

## Function and method of operation

Block OB_BEGIN is used to report and display CPU events and statuses. It reads the start information of the tasks (OBs), diagnostic data of the I/O and enables the relevant blocks for processing on the basis of the start events.

OB_BEGIN reads the start information of SFC 6 (RD_SINFO) or SFB 54 (RALRM) to identify the OB in which it is currently running.

If this information is not available, the block reads the logical basic address from the start information and converts it into the geographic address. This is available at the relevant OBs of the output structure CPU_DIAG the lower-level blocks can also access. On the basis of the geographic address, OB_BEGIN enables the relevant SUBNET blocks for further evaluation of the start information.

In the case of a diagnostic event (OB 82), the diagnostic data are simultaneously written to the CPU_DIAG.OB82 structure along with the start information, using SFB54. Diagnostic (alarm) information of a length exceeding 59 bytes must contain the call of the relevant driver blocks.

In H systems, the current status of the two H CPUs is read from the system status list 71 (SSL71) in OB 100. A detailed description of the SSL71 appears in the reference manual titled *System Software for S7-300/400 System and Standard Functions*. The statuses of SSL_71.MASTER_0/1 and SSL_71.CPU_ERR_0/1 are updated in OB 72.

OB_BEGIN provides the diagnostic information of OB 55, OB 56 and OB 57 to the downstream blocks in its output structure CPU_OB_5X.

The block reports diagnostic events of an OB 88 block.

All OB 88 events are incoming events. OB 1 generates the relevant event message exiting state after a delay of approximately 10 seconds in order to allow the reporting of a new OB 88 event.

## Error handling

Block OB_BEGIN evaluates error information from ALARM_8P and writes it to the relevant output parameters.

You will find additional information in the "Error information of output parameter MSG_STAT (Page 398)" section.

If the block installation sequence OB_BEGIN, xx blocks, ..., OB_END is not observed in an OB, the message "OB_END installation error, no OB 8x processing" will be output and QERR = TRUE set. In this case, the acyclic OBs do not evaluate the data. The downstream blocks will not be enabled.

Error information at output parameter STATUS of SFB 54 (RALRM) is handled as follows:

- The values 16#8096, 16#80A7, 16#80C0, 16#80C2, 16#80C3 or 16#80C4 at STATUS[2] and STATUS[3] indicate temporary errors. STATUS[3] of the corresponding OB will be set in the structure CPU_DIAG = 16#C4. Downstream blocks can read access the diagnostic data asynchronously.
- After any other error event, SFC 6 (RD_SINFO) reads the startup information once again and the message "OB_BEGIN diagnostic error RALRM STATUS = xxxxxxxx" is output. OB 1 generates the message exiting state once a delay of approximately 10 seconds has expired.

## Startup characteristics

Block OB_BEGIN initializes the messages of ALARM_8P. In H systems (CPU_DIAG.H_MODE = TRUE), the current status of the two H CPUs is determined by reading SSL71 (see "Function and method of operation").

## Overload Behavior

Messages exiting state associated with OB 121, OB 122 and OB 88 are generated subject to a delay of approx. 10 seconds. This on the one hand prevents blocking of the WinCC connection due to a high message transfer volume of these OBs. On the other hand, OB events may be due to the delay.

## Time response

Not available

## Message response

ALARM_8P multiple instances are only called if OB_BEGIN is to output a message. It is only at this point that previously acknowledged messages are updated by the corresponding ALARM block. If the connection to WinCC is down, each ALARM_8P can hold up to two message statuses of its event ID.

The CPU generates a programming error (OB 121) only as an incoming event. OB 1 resets the relevant message to status exiting state. In order to avoid an excessive number of programming error messages, these will not be reported as outgoing until a delay time of 10 seconds has expired. The same applies to I/O access errors (OB 122) and OB 88 events.

The block generates the following messages in the OBs listed below:

| OB | Start Event | Message |
|---|---|---|
| OB 1 | Cyclic processing | • Outgoing message with 10 s delay: Timeout (OB 80/OB 84)<br>• Program execution error (OB 80)<br>• Programming error (OB121)<br>• Write I/O access error (OB 122)<br>• Read I/O access error (OB 122)<br>• Error code B#16#71: Nested stack error (OB 88)<br>• Error code B#16#72: Master control relay stack error (OB 88)<br>• Error code B#16#73: Synchronous error nesting depth exceeded (OB 88)<br>• Error code B#16#74: U stack nesting depth exceeded in priority class stack (OB 88)<br>• Error code B#16#75: B stack nesting depth exceeded in priority class stack (OB 88)<br>• Error code B#16#76: Local data allocation error (OB 88)<br>• Error code B#16#78: Unknown opcode (OB 88)<br>• Error code B#16#7A: Code length error (OB 88) |
| OB72 | CPU redundancy loss | CPU redundancy loss/return |
| OB 80 | Timeout error | Incoming message on timeout:<br>• Cycle time exceeded<br>• OB request: OBxx is busy<br>• OB request: Overflow PRIOxx<br>• TOD interrupt xx expired |
| OB 84 | CPU hardware error | Interface error entering/exiting state;<br>• Memory error detected and corrected by operating system.<br>• Accumulation of detected and corrected memory errors.<br>• Error in PC operating system.<br>• Performance of an H-Sync coupling impaired.<br>• Multiple-bit memory error detected and corrected. |
| OB 85 | Program execution error | Incoming message on program-execution error:<br>• OBxx not loaded<br>• Access-error error xx: ... |

| OB | Start Event | Message |
|---|---|---|
| OB 88 | Stop avoidance | Incoming message on OB 88 events:<br>• Error code B#16#71: Nested stack error<br>• Error code B#16#72: Master control relay stack error<br>• Error code B#16#73: Nesting depth exceeded on synchronization errors<br>• Error code B#16#74: U-stack nesting depth exceeded in the priority class stack<br>• Error code B#16#75: B-stack nesting depth exceeded in the priority class stack<br>• Error code B#16#76: Local data allocation error<br>• Error code B#16#78: Unknown opcode<br>• Error code B#16#7A: Code length error |
| OB100 | Restart | Initialization of ALARM_8P |
| OB121 | Programming error | Programming error incoming |
| OB122 | I/O access error | • Read I/O access, incoming<br>• Write I/O access, entering state |

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

- OB_BEGIN faceplate

If no asset management is used in the project, the "OB-BEGIN" block icon is used to display avoidance of stop.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of OB_BEGIN (Page 224)

Maintenance status of MS (Page 403)

## 3.29.2 I/Os of OB_BEGIN

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number ALARM_8P_x<br>(x = 1 - 4, assigned by the ES) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| SUB0IDxx | DP master system 1 IDxx (xx = 00 - 14) | BYTE | 255 |
| SUB1IDxx | DP master system 2 IDxx (xx = 00 - 14) | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| **CPU_DIAG** | System structure: CPU diagnostics | STRUCT | |
| CPU_OB_4X | Start information OB 40 - OB 47 | STRUCT | |
| CPU_OB_5X | Start information OB 55, OB 56, OB 57 | STRUCT | |
| CPU_OB_6X | Start information OB 60 - OB 64 | STRUCT | |
| **CPUERR_0** | 1 = CPU error in rack 0 *) | BOOL | 0 |
| **CPUERR_1** | 1 = CPU error in rack 1 *) | BOOL | 0 |
| EN_SUBx | Enable SUBNET x (x = 0 - 14) | BOOL | 0 |
| **MASTER_0** | 1 = Master CPU in rack 0 | BOOL | 0 |
| **MASTER_1** | 1 = Master CPU in rack 1 | BOOL | 0 |
| MSGSTATx | STATUS output of ALARM_8P_x (x = 1 -4) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QERR | 1 = processing error | BOOL | 1 |
| QMSGERx | Error output of ALARM_8P_x (x = 1 - 4) | BOOL | 0 |
| SZL_71 | System structure: SZL71 | STRUCT | |

The structure of the CPU_DIAG is integrated as OUT in the OB_BEGIN block, and as IN_OUT in all other blocks with this I/O.

*) You will find additional information about CPU errors in the CPU Manual.

## Additional information

You will find more information in:

Message texts and associated values of OB_BEGIN (Page 224)

Maintenance status of MS (Page 403)

## 3.29.3 Message texts and associated values of OB_BEGIN

### Assignment of message text and message class

Process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message block ALARM_8P | Message number | OB no. | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 | 1 | OB 85 | OB @7%d@ not loaded | S |
| | 2 | | | No message |
| | 3 | OB 84 | Interface error | S |
| | 4 | - | Error installing OB_BEGIN/OB_END: No OB@2%d@ processing of stack @1%d@ | S |
| | 5 | OB 85 | Program execution error: @7%d@: @10%2s@@8%d@/@9%d@ | S |
| | 6 | OB 122 | I/O read access error: @4%2s@@5%d@ Address: @6%d@ | S |
| | 7 | OB 122 | I/O write access error: @4%2s@@5%d@ Address: @6%d@ | S |
| | 8 | OB 84 | Performance of an H-Sync coupling impaired | S |

Messages 1, 4, 5, 6, 7 and 8 are only incoming events. They are reset to "outgoing" status during the normal run sequence (OB 1) of the block.

### Associated Values of ALARM_8P with EV_ID1

Process control messages are generated with six associated values at EV_ID1 via ALARM_8P. The table below shows how the associated values are assigned to the block parameters.

| Message block ALARM_8P | Associated value | Block parameters | Data type |
|---|---|---|---|
| EV_ID1 | 1 | CPU_DIAG.OB_S_NUM_CNT | BYTE |
| | 2 | TINFO_TOP_SI_NUM | BYTE |
| | 3 | OB 72_supp_info 1 | WORD |
| | 4 | OB 122_BLK_TYP | WORD |
| | 5 | OB 122_BLK_NUM | WORD |
| | 6 | OB 122_MEM_ADDR | WORD |
| | 7 | OB 85_supp_info 1 | WORD |
| | 8 | OB 85_HW_supp_info 2_3 | WORD |
| | 9 | OB 85_LW_supp_info 2_3 | WORD |
| | 10 | OB 85_DKZ2_3 | WORD |

## Process control messages of ALARM_8P with EV_ID2 are assigned as follows:

| Message block ALARM_8P | Message number | OB no. | Default message text | Message class |
|---|---|---|---|---|
| EV_ID2 | 1 | OB 80 | Net consumption of all OBs exceeds max. limit | M |
| | 2 | OB 80 | Emergency operation, cyclic OBs are used | S |
| | 3 | OB 80 | Priorities of the cyclic OBs not PCS 7 conform | M |
| | 4 | OB 84 | Memory error detected and corrected by operating system. | S |
| | 5 | OB 84 | Accumulation of detected and corrected memory errors | S |
| | 6 | OB 84 | Error in PC operating system | S |
| | 7 | OB 121 | Programming error @1%d@: @2%2s@@5%d@ /@6%d@/@4%d@/@3%d@ | S |
| | 8 | OB 84 | Multiple-bit memory error detected and corrected | S |

Messages 1 to 3 are generated in CPU_RT (Page 23) and forwarded to OB_BEGIN.

Messages 4, 5, 7 and 8 are only incoming events. They are reset to "outgoing" status during the normal run sequence (OB 1) of the block.

Message 7 is to be interpreted as follows, in accordance with the error code number before the colon:

OB 121_BLK_TYP/OB 121_BLK_NUM/OB 121_PRG_ADDR/OB 121_FLT_REG/OB 121_RESERVED_1.

This is described in the reference manual titled *System Software for S7-300/400 System and Standard Functions*.
Example: 10.05.00 10:30:45 Programming error 35: FB44/1234/5/9

## Associated Values of ALARM_8P with EV_ID2

Process control messages are generated with six associated values at EV_ID2 via ALARM_8P. The table below shows how the associated values are assigned to the block parameters.

| Message block ALARM_8P | Associated value | Block parameters | Data type |
|---|---|---|---|
| EV_ID2 | 1 | OB 121_SW_FLT | BYTE |
| | 2 | OB 121_BLK_TYP | WORD |
| | 3 | OB 121_RESERVED_1 | BYTE |
| | 4 | OB 121_FLT_REG | WORD |
| | 5 | OB 121_BLK_NUM | WORD |
| | 6 | OB 121_PRG_ADDR | WORD |

## Process control messages of ALARM_8P with EV_ID3 are assigned as follows:

| Message block ALARM_8P | Message number | OB no. | Default message text | Message class |
|---|---|---|---|---|
| EV_ID3 | 1 | OB 80 | Cycle time exceeded: @1%d@ms OB@2%d@ | S |
| | 2 | OB 80 | OB request: OB3x still being processed | F |
| | 3 | OB 80 | TOD interrupt OB @3%d@ expired (TOD jump) | S |
| | 4 | OB 80 | TOD interrupt OB @4%d@ expired (Stop/Run) | S |
| | 5 | OB 80 | OB request: Overflow PRIO @5%d | S |
| | 6 | OB 80 | Clocked interrupt timeout: OB@6%d@ PRIO @7%d@ | S |
| | 7 | OB 80 | Interrupt lost: OB@8%d@ PRIO @9%d@ | S |
| | 8 | OB 80 | CiR synchronization time: @10%d@ ms | S |

Message 2 is generated in CPU_RT (Page 23) and forwarded to OB_BEGIN.

Messages 1 to 8 are only incoming events. They are reset to "outgoing" status during the normal run sequence (OB 1) of the block.

## Associated Values of ALARM_8P with EV_ID3

Process control messages are generated with seven associated values at EV_ID3 via ALARM_8P. The table below shows how the associated values are assigned to the block parameters.

| Message block ALARM_8P | Associated value | Block parameters | Data type |
|---|---|---|---|
| EV_ID3 | 1 | Cycle time (OB 80_supp_info 1) | WORD |
| | 2 | Cause OB (OB 80_1st byte supp_info 2_3) | BYTE |
| | 3 | Cycle time (OB 80_supp_info 1) | WORD |
| | 4 | Cycle time (OB 80_supp_info 1) | WORD |
| | 5 | Priority class (OB 80_2nd byte supp_info 2_3) | BYTE |
| | 6 | Cause OB (OB 80_1st byte supp_info 2_3) | BYTE |
| | 7 | Priority class (OB 80_2nd byte supp_info 2_3) | BYTE |
| | 8 | Cause OB (OB 80_1st byte supp_info 2_3) | BYTE |
| | 9 | Priority class (OB 80_2nd byte supp_info 2_3) | BYTE |
| | 10 | Cycle time (OB 80_supp_info 1) | WORD |

## Process control messages for ALARM_8P with EV_ID4 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID_4 | 1 | OB 88(@6W%t#OB_BEGIN_TXT@): OB@1%d@ PRIO@2%d@ @3%2s@@4%d@ /@5%d@ | S |
| | 2 | OB_BEGIN: Diagnostics error RALRM STATUS = @7%8X@ | S |
| | 3 | | No message |
| | 4 | | No message |
| | 5 | | No message |
| | 6 | | No message |
| | 7 | | No message |
| | 8 | | No message |

## Associated Values of ALARM_8P with EV_ID4

| Message block ALARM_8P | Associated value | Meaning |
|---|---|---|
| EV_ID4 | 1 | Cause OB (M_OB 88.FLT_OB) |
| | 2 | Priority class (M_OB 88.FLT_OB_PRIO) |
| | 3 | Block type (M_OB 88.BLK_TYP) |
| | 4 | Block number (M_OB 88.FLT_NUM) |
| | 5 | MC7 command causing error<br>Relative address (M_OB 88.FLT_ADDR) |
| | 6 | Error number in OB_BEGIN_TXT (M_OB 88.T_OB 88) |
| | 7 | Status RALRM |

You will find the message texts and their text numbers in "Text library for OB_BEGIN (Page 426)".

## See also

Message Classes (Page 401)

# 3.30 OB_DIAG1: OB diagnostics for avoiding stoppages in DPV1 master systems

## 3.30.1 Description of OB_DIAG1

### Object name (type + number)

FB 118

- OB_DIAG1 block I/Os (Page 232)

### Area of application

Block OB_DIAG1 monitors the failure and recovery of DP or PA slaves (referred to as "slaves" below). The slaves can be connected to a DPV0 or DPV1 master system, or to a DPV1 DP/PA link (Y link). OB_DIAG1 blocks further evaluation if a slave is defective (frequent producer) to prevent the CPU stopping. It indicates the preferred channel of the active slave in an H system. The indicated preferred channel 1 (SUBN1ACT ) is always TRUE, if the slave is downstream of a DP/PA link (Y link) and is active.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 70 | I/O redundancy error |
| OB 72 | CPU redundancy error |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |
| OB 55 | Status interrupt (only as required) |
| OB 56 | Update interrupt (only as required) |
| OB 57 | Manufacturer-specific alarms (only as required) |

The driver generator only installs the block in OB 55, OB 56 and OB 57 if diagnostic messages are to be expected from these locations; consequently OB 5x are not entered in this block's task list.

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- Block OB_DIAG1 is installed downstream of the SUBNET/DPAY_V1 block (when used downstream of a DP/PA or Y link).
- The RACK_NO, LADDR, DADDR, EN_MSG_D, SUBN1_ID, SUBN2_ID, and SUBN_TYP inputs are configured.
- The EN input is interconnected with the output of an AND block.
  The block inputs will be interconnected with output EN_SUBx (x = DP master system ID) of the OB_BEGIN block, and with output EN_Rxxx (xxx = rack/station number) of the SUBNET block.
- The CPU_DIAG OUT structure of the OB_BEGIN block and SUB_DIAG of the SUBNET block are interconnected with the IN_OUT structures of the same name of block OB_DIAG.
- RAC_DIAG_I is interconnected with its own RAC_DIAG OUT structure when used in a DP master system.
- RAC_DIAG_I is interconnected with the RAC_DIAG OUT structure of the DPAY_V1 block when used downstream of a Y link.

## Function and method of operation

The DPA_LINK block input receives information on the operational status of the slave, in other words whether it is in use on a DP master system (DPA_LINK = FALSE) or downstream of a DP/PA link (Y link). If the slave is connected to a DP master system, a failure is reported in OB 86. If the slave is connected downstream of a DP/PA link (Y link), the failure is reported in OB 83.

The block counts the number of calls to an acyclic OB of a block instance before an OB 1 can be executed.

OB_DIAG1 indicates higher-level errors of the CPU, DP master/slave (QRACKF, SUBN1ERR, SUBN2ERR). It determines the preferred channel of connected DP slaves (SUBN1ACT, SUBN2ACT). The group error message QRACKF indicates failure of the DP master or slave. An active DP slave has lost redundancy if one of the output parameters SUBN1ERR or SUBN2ERR = FALSE.

Start and diagnostic information is read from the CPU_DIAG structure, which is interconnected with the CPU_DIAG structure of the OB_BEGIN block.

The block evaluates error events, and uses the diagnostic address DADDR of the slave (on the DP master system only) to determine the currently active preferred channel (SUBN1ACT, SUBN2ACT) of redundant PROFIBUS DP interface circuits.

The slaves downstream of a DP/PA link (Y link) are not always active. In this case, the diagnostic address DADDR is the diagnostic address of the link. The active preferred channel (SUBN1ACT, SUBN2ACT) is displayed here by the DP/PA link (Y link).

The DP master systems or DP/PA links (Y link) must be operated in DPV1 mode (V1-MODE = TRUE).

Failure and return of a DP slave are reported with ALARM_8P. The message function for all messages can be disabled with EN_MSG = FALSE.

The “Device failure” message can be disabled with EM_MSG_D = FALSE (see “Message Response”).

## Overload behavior

Block OB_DIAG1 counts the frequency of the calls to the acyclic OB 55, OB 56, OB 57, OB 82, and OB 86 blocks (except in the case of a DP master system failure, see SUBNET block). If the block is downstream of a DP/PA or Y link, the calls will be counted in OB 83, rather than in OB 86. The following section deals only with OB 86.

Each OB is assigned a counter that is checked for the condition > 5. If this condition is fulfilled, the block sets EN_F = FALSE (disable function block). The counters are reset in OB 1. The output EN_F = TRUE (enable function block) is set in all other OBs.

OB_DIAG1 reports failure of the blocks mentioned above in OB 1, OB 82 or OB 86, including the geographic address of the slave.

OB 55, OB 56, OB 57 and OB 82 are locked in the event of an overload, however, so the event is not evaluated in the downstream blocks. The outputs cannot correspond to the current slave status. If an OB is locked and no more slave events have been reported after a delay of around 1 minute, if it is OB 86 that is disabled, the slave status is checked, and the outputs are updated. It may take several cycles to update the slave status.

If it is OB 82 that is disabled, rather than OB 86, the EN_DIAG variable is set to TRUE after around 1 minute. The interconnected DP slave block can then fetch the current diagnostic data for the slave, and update its own data. The same applies to OB 55, OB 56, and OB 57.

The “outgoing” message about the fault is generated when the OB lock is canceled, and either a new event has occurred for this OB or the wait time has elapsed.

## Redundancy

The block supports redundant DP master systems in an H system (distributed I/Os only). The SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs of the OB_DIAG1 block are configured with the numbers of the redundant DP master systems. If the DP master systems are not redundant, the remaining input is set to 16#FF (default).

## Startup characteristics

The availability of the slave is checked. In H systems the preferred channel of the slave is determined (active slaves only).

## Error handling

The block evaluates the error information from ALARM_8P, and writes it to the relevant output parameter.

You will find additional information in the "Error information of output parameter MSG_STAT (Page 398)" section.

## Message response

The multiple instances ALARM_8P are only called if a message is to be output by this instance. It is only at this point that previously acknowledged messages are updated by the corresponding ALARM block. If the connection to WinCC is down, each ALARM_8P instance can hold up to two message statuses of its event ID. (and generally no more than two messages). The block generates the messages listed below:

| OB no. | Start Event | Message |
|---|---|---|
| 1 | Cyclic processing | Call of ALARM_8P due to incomplete transfer or unacknowledged message |
| 72 | CPU redundancy loss | If no redundant diagnostic repeater is connected to this CPU, the "Slave" failure/return message is output. |
| 70 | Redundancy loss | If there is no redundant slave connected to this DP master system, message "Device" failure/return, otherwise message "Slave" redundancy loss/return |
| 83 | Removal/insertion | Message "Slave" failure/return |
| 86 | Rack failure | Message "Slave" failure/return |
| 100 | Restart | Initialization of ALARM_8P |

If the diagnostic blocks (e.g., MOD_PAL0) of a device also report the failure of a device, the "Device failure“ message can be disabled with EN_MSG_D = FALSE (this is done automatically by the driver generator).

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of OB_DIAG1 (Page 234)

Maintenance status of MS (Page 403)

## 3.30.2 I/Os of OB_DIAG1

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **DADDR** | Diagnostic address of the slave or DP/PA link | INT | 0 |
| **DPA_LINK** | Slave connection: 0 = DP master system 1 = Link | BOOL | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EN_MSG_D | 1 = Enable message "Device failure" | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **LADDR** | Logical basic address of the slave | INT | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **PADP_ADR** | Address of the PA/DP slave | BYTE | 255 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number 0 of the slave at the DP/PA Link | BYTE | 255 |
| **SUBN_TYP** | 1 = External DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **EN_DIAG** | 1 = Read diagnostics with SFC 13 | BOOL | 0 |
| **EN_F** | 1 = Enable function/function block | BOOL | 0 |
| MOD_INF | System structure: Module diagnostics | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| **QRACKF** | 1 = Slave failure/faulty | BOOL | 0 |
| RAC_DIAG | System structure: RACK diagnostics | STRUCT | |
| **SUBN1ACT** | 1 = Slave 1 is active | BOOL | 0 |
| **SUBN1ERR** | 1 = Error in the primary DP master system | BOOL | 0 |
| **SUBN2ACT** | 1 = Slave 2 is active | BOOL | 0 |
| **SUBN2ERR** | 1 = Error in the redundant DP master system | BOOL | 0 |
| **V1_MODE** | 1 = DPV1 mode of the DP master system | BOOL | 0 |

### In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_ID | 1 = Accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| CPU_OB_5X | OB_5x start information | STRUCT | |
| RAC_DIAG_I | System structure: RACK diagnostics | STRUCT | |
| SUB_DIAG | System structure: SUBNET diagnostics | STRUCT | |

### Additional information

For additional information, refer to the following sections:

Message texts and associated values of OB_DIAG1 (Page 234)

Maintenance status of MS (Page 403)

### 3.30.3 Message texts and associated values of OB_DIAG1

#### Assignment of message text and message class

| Message No. | Default message text | Message class |
|---|---|---|
| 1 | DP slave @4%d@/ @2%d@: Redundancy loss | F |
| 2 | DP slave @1%d@/ @2%d@: Failure | S |
| 3 | DP slave @1%d@/ @2%d@/@3%d@ : Multiple failure | S |
| 4 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 82) | S |
| 5 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 55) | S |
| 6 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 56) | S |
| 7 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 57) | S |
| 8 | Device @1%d@/ @2%d@/@3%d@: Failure | S |

#### Assignment of Associated Values

| Associated value | Block parameters |
|---|---|
| 1 | ID of the primary DP master system (SUBN_ID) |
| 2 | Rack/station number (RACK_NO) |
| 3 | Slot number (SLOT_NO) |
| 4 | Subnet ID of the master in the event of redundancy loss (SUBN2_ID) |

#### See also

Message Classes (Page 401)

## 3.31 OB_DIAGF: Monitoring of the FF link and coupler

### 3.31.1 Description of OB_DIAGF

#### Object name (type + number)

FB 146

- I/Os of OB_DIAGF (Page 237)

#### Area of application

The OB_DIAGF block monitors the FF link and coupler for

- Failure and return of FF link and coupler
- Redundancy loss
- diagnostic interrupts

and reports these events.

The OB_DIAGF block runs acyclically and is enabled to run by the higher-level SUBNET block.

#### Calling OBs

The block must be installed in the run sequence downstream from the SUBNET block in the following OBs (this is done automatically in the CFC):

| OB no. | Start Event |
|---|---|
| 1 | Cyclic processing |
| 72 | CPU redundancy loss |
| 70 | Redundancy loss |
| 82 | Diagnostic interrupt |
| 83 | Removal/insertion |
| 85 | Program execution error |
| 86 | Module driver failure |
| 100 | Restart |

## Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The OB_DIAGF block is installed downstream from the SUBNET block.
- The DADDR, SUBN1_ID, SUBN2_ID, RACK_NO, COUP_RED, and SLOT_NO inputs are assigned parameters.
- The EN input is interconnected with the output of an AND block. The block inputs will be interconnected with output EN_SUBx (x = DP master system ID) of the OB_BEGIN block and with output EN_Rxxx (xxx = rack/station number) of the SUBNET block.
- The CPU_DIAG OUT structure of the OB_BEGIN block and SUB_DIAG of the SUBNET block are interconnected with the IN_OUT structures of the same name of block OB_DIAGF.

## Message response

The DELAY input is used to delay the outputting of error messages for an outgoing, higher-priority error. For example, if the OB_DIAGF block recognizes an outgoing error at a DP master connected with it, it initially assumes that there is a faulty assigned DP slave in the rack it monitors, and sets the corresponding output SUBNxERR. The error status is not reset until the DP slave returns (in this case: OB 86, OB 70). The OB_DIAGF blocks suppress the potential slave failure for DELAY seconds so as not to trigger a surge of messages from DP slaves which are not yet synchronized when the master returns. An error message is only output to the OS when the DP slave has not reported its return before this delay time has expired.

Note: Do not set the value of DELAY too high, otherwise DP slaves that were removed during the master failure or are defective will be signaled to the OS too late after the DP master returns.

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## 3.31.2 I/Os of OB_DIAGF

### I/Os of OB_DIAGF

The factory setting of the block display in CFC is identified in the "I/O" column.

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **COUP_RED** | 0 = No redundant structure<br>1 = Coupler redundancy<br>2 = Ring redundancy | BYTE | 16#FF |
| **DADDR** | Diagnostic address of the FF link | INT | 0 |
| DELAY | Interrupt delay (s) | INT | 15 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID1 | Message number | DWORD | 0 |
| EV_ID2 | Message number | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Number of the rack | BYTE | 16#FF |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 16#FF |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 16#FF |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DINFO | Diagnostic information | STRUCT | |
| EN_F | 1 = Enable function/function block | BOOL | 0 |
| FF_DIAG | Diagnostics of the FF device | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QRACKF | 1 = Slave failure/faulty | BOOL | 0 |
| RAC_DIAG | RACK diagnostics | STRUCT | |
| RETURN_CODE_O | Return value | INT | 0 |
| SUBN1ACT | 1 = Slave 1 is active | BOOL | 0 |
| SUBN2ACT | 1 = Slave 2 is active | BOOL | 0 |
| SUBN1ERR | 1 = Error in the primary DP master system | BOOL | 0 |
| SUBN2ERR | 1 = Error in the redundant DP master system | BOOL | 0 |
| V1_MODE | 1 = DPV1 mode of the DP master system active | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| SUB_DIAG | System structure: SUBNET diagnostics | STRUCT | |

## 3.31.3 Message texts and associated values of OB_DIAGF

### Message texts and associated values of OB_DIAGF

Assignment of message text and message class (Page 401)

| Message block ALARM_8P | Message No. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | FF link @4%d@/@2%d@:Redundancy loss | F |
| | 2 | FF link @1%d@/@2%d@: Failure | S |
| | 3 | FF link @1%d@/@2%d@: Multiple failure | S |
| | 4 | FF link @1%d@/@2%d@: Multiple alarm (OB 82) | S |
| | 5 | Link @1%d@/@2%d@/2: Redundancy loss | F |
| | 6 | Link @1%d@/@2%d@/3: Redundancy loss | F |
| | 7 | FF-Link @1%d@/@2%d@/1:DIP switch has a different DP address setting than at "power on" of the IM/Redundant configuration, neighboring IM has a different DP address. | S |
| | 8 | FF link @1%d@/@2%d@/1: SDB_UUID does not match saved UUID | S |
| EV_ID2 | 1 | Link @1%d@/@2%d@/2: Failure | S |
| | 2 | Link @1%d@/@2%d@/2: Contrary to configuration: Coupler configuration does not match the determined configuration | S |
| | 3 | Link @1%d@/@2%d@/2: External connection, short circuit or wire break detected | S |
| | 4 | Link @1%d@/@2%d@/2: Loss of redundancy in the case of ring redundancy | F |
| | 5 | Link @1%d@/@2%d@/3: Failure | S |
| | 6 | Link @1%d@/@2%d@/3: Contrary to configuration: Coupler configuration does not match the determined configuration | S |
| | 7 | Link @1%d@/@2%d@/3: External connection, short circuit or wire break detected | S |
| | 8 | Link @1%d@/@2%d@/3: Loss of redundancy in the case of ring redundancy | F |

### Assignment of Associated Values

| Associated value | Block parameters |
|---|---|
| 1 | DP master system ID (SUBN_ID) |
| 2 | Rack/station number (RACK_NO) |

# 3.32 OB_END: Reset stack pointer of OB_BEGIN

## 3.32.1 Description of OB_END

### Object name (type + number)

FC 280

- OB_END block I/Os (Page 242)

### Area of application

The OB_END block is used to reset the stack pointer of OB_BEGIN and OB_BEGIN_PN.

### Calling OBs

The OB_END block is the final entry in the OB that contains an OB_BEGIN and/or an OB_BEGIN_PN block.

| | |
|---|---|
| OB 1 | Cyclic processing |
| OB 55 | Status interrupt (only as required) |
| OB 56 | Update interrupt (only as required) |
| OB 57 | Manufacturer-specific alarms (only as required) |
| OB 70 | I/O redundancy error |
| OB 72 | CPU redundancy error |
| OB 80 | Timeout error |
| OB 81 | Power supply error |
| OB 82 | Diagnostic interrupt |
| OB 83 | Remove/insert interrupt |
| OB 84 | CPU hardware error (only for CPU with this function) |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 88 | Stop avoidance |
| OB 100 | Restart |
| OB 121 | Programming error |
| OB 122 | I/O access error |

### Use in CFC

When the CFC function "**Generate module drivers**" is used, OB_END is automatically installed in the above OBs of the run sequence.

## Function

The OB_END block decrements the stack pointer (NUM_CNT) of OB_BEGIN and OB_BEGIN_PN. In the event of an interruption, it enters the last interrupted OB number read from the CPU stack into the CPU_DIAG and CPU_DIAG_PN structure.

## Error handling

Not available

## Startup characteristics

Not available

## Initial startup behavior

Not available

## Time response

Not available

## Message response

Not available

## Operator control and monitoring

The block has no faceplate.

## 3.32.2 I/Os of OB_END

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### In-out parameters

| I/O (parameter) | Meaning | Data type |
|---|---|---|
| **CPU_DIAG** | System structure: CPU diagnostics | STRUCT |
| **CPU_DIAG_PN** | System structure: CPU diagnostics (PN) | STRUCT |

# 3.33 OR_32_TS: OR value status of two redundant time-stamped signal modules, max. 32 channels

## 3.33.1 Description of OR_32_TS

### Object name (type + number)

FB 138

- OR_32_TS block I/Os (Page 245)

### Area of application

The OR_32_TS block forms the resulting time stamp from two redundant time-stamped signal modules.

### Calling OBs

The block must be installed in OB 1.

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The OR_32_TS block is installed in OB 1.
- The `TS1_xx` inputs are interconnected with the `TS_xx` output of IMDRV_TS that is responsible for the signal module with the lower address.
- The `TS2_xx` inputs are interconnected with the `TS_xx` output of IMDRV_TS that is responsible for the signal module with the higher address.
- The `TS_xx` outputs are interconnected with the inputs of the MSG_TS channel blocks or Pcs7InIT.

### Function and method of operation

The OR_32_TS block forwards the time stamps of the channels of two redundant signal modules to the MSG_TS channel blocks or Pcs7InIT and acts like an OR function.

- If both channels are active, the time stamp of the signal module with the lower address is always used.
- If one channel is passivated, the time stamp of the redundant channel is forwarded.
- If both channels are passivated, the time stamp of the signal module with the lower address is entered.

## Redundancy

Redundancy of the modules in an H system is monitored in the higher-level RED_STATUS block.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

Not available

## Message response

Not available

## Operator control and monitoring

Not available

## Additional information

You will find more information in Description of IMDRV_TS (Page 89)

## 3.33.2 I/Os of OR_32_TS

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| **CH_INF_H** | Status of the channels of the 2nd module; information channel by channel<br>0 = passivated, 1 = operational | DWORD | 0 |
| **CH_INF_L** | Status of the channels of the 1st module; information channel by channel<br>0 = passivated, 1 = operational | DWORD | 0 |
| RED | 1 = with redundant time stamp | BOOL | 0 |
| RED_STAT | Return value from RED_STATUS | INT | 0 |
| TS1_xx | Time stamp (xx = 00 - 31) from the module with the lower address<br>Byte 0:<br>Bit 0: Message signal state (MsgSig)<br>Bit 1: Edge change information (TriInf)<br>Bit 2: Handshake (HdSh)<br>Byte 1:<br>Quality code of the time stamp (ST)<br>DWORD TS0:<br>Date/time stamp in ISP format (seconds)<br>DWORD TS1:<br>Date/time stamp in ISP format (fractions of seconds) | STRUCT | |
| TS2_xx | Time stamp (xx = 00 - 31) from the module with the higher address<br>Byte 0:<br>Bit 0: Message signal state (MsgSig)<br>Bit 1: Edge change information (TriInf)<br>Bit 2: Handshake (HdSh)<br>Byte 1:<br>Quality code of the time stamp (ST)<br>DWORD TS0:<br>Date/time stamp in ISP format (seconds)<br>DWORD TS1:<br>Date/time stamp in ISP format (fractions of seconds) | STRUCT | |

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| TS1_C_xx | TS communication (xx = 00 - 31) from the module with the lower address<br>Bit 0: Acknowledgment of transfer (HS)<br>Bit 1: Interconnection check (LI) | BYTE | 0 |
| TS2_C_xx | TS communication (xx = 00 - 31) from the module with the higher address<br>Bit 0: Acknowledgment of transfer (HS)<br>Bit 1: Interconnection check (LI) | BYTE | 0 |

## Output parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| CH_ALM | Channel failure Redundant pair | DWORD | 0 |
| CH_WRN | Channel redundancy loss | DWORD | 0 |
| QERR | 1 = program error<br>(module status not available) | BOOL | 1 |
| TS_xx | Time stamp (xx = 00 - 31)<br>Byte 0:<br>Bit 0: Message signal state (MsgSig)<br>Bit 1: Edge change information (TriInf)<br>Bit 2: Handshake (HdSh)<br>Byte 1:<br>Quality code of the time stamp (ST)<br>DWORD TS0:<br>Date/time stamp in ISP format (seconds)<br>DWORD TS1:<br>Date/time stamp in ISP format (fractions of seconds) | STRUCT | |
| TS_C_xx | TS communication (xx = 00 - 31)<br>Bit 0: Acknowledgment of transfer (HS)<br>Bit 1: Interconnection check (LI) | BYTE | 0 |

## Additional information

For additional information, refer to the following sections:

- Maintenance Status of MS (Page 403)

# 3.34 OR_HA16C: OR value status of 2 redundant HART modules, max. 16 channels, module granular

## 3.34.1 Description of OR_HA16C

### Object name (type + number)

FB 133

- OR_HA16C block I/Os (Page 250)

### Area of application

The OR_HA16C block is used to create a value status from two redundant signal modules, and reports loss of redundancy for HART modules.

### Calling OBs

The block must be installed in OB 100 and in the OB before the MOD_HA driver block that is responsible for the relevant module.

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The OR_HA16C block is installed before its interconnected MOD_HA driver block in its OB.
- MODE1_xx inputs are interconnected with the OMODE_xx outputs of the MOD_x block in the primary module.
- MODE2_xx inputs are interconnected with the OMODE_xx outputs of the MOD_x block in the redundant module.
- The MOD_INF1 input structure is interconnected with the MOD_INF output structure of the MOD_x block in the primary module.
- The MOD_INF2 input structure is interconnected with the MOD_INF output structure of the MOD_x block in the redundant module.
- The ACTIV_H and ACTIV_L inputs are interconnected with the outputs of the same name of the RED_STATUS block in the redundant module.
- The OMODE_xx outputs are interconnected with the downstream MOD_HA.
- The OUT structure CPU_DIAG of the OB_BEGIN block is interconnected with the IN_OUT structures of the same name of the OR_HA_16C block.
- The inputs RACKF1 and RACKF2 are interconnected with the outputs QRACKF1 and QRACKF2 of MOD_x.

- The CH_INF_H and CH_INF_L inputs are interconnected with the outputs of the same name of the RED_STATUS block.
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function and method of operation

The value status of a signal module and of a redundant signal module are ORed by the OR_HA16C block. Signal modules or signal channels set to passive mode by the system are regarded as invalid. The "At least one channel pair does not match", "Redundancy loss" and "Failure of the redundant I/O" events are reported by ALARM_8P. The message function can be disabled.

When module signals are processed in different OB 3x, in very few cases, a faulty signal value may be processed in one cycle by one of the channel blocks configured to process the module signals following a process-control malfunction affecting the module. This can be avoided by processing all channel blocks of a module in the OB3x that also processes the process image partition to which this module is assigned.

For a passivated block or passivated channel when setting the DEPASS input the depassivation can be triggered. For this the function RED_DEPA (FC 451) is called internally.

**For OR_M_xxC / OR_HA16C blocks:**

For channel-granular redundancy, in the event of one channel failing, "Loss of Redundancy Channel x" is reported. If both channels fail, "Redundancy pair failure channel x" is reported.

With OR_HA16C, the message "Redundancy pair failure HART variable 01" to "Redundancy pair failure HART variable 08" is output for HART signals 01 to 08. The message function can be disabled.

## For all OR blocks

Depassivation can be triggered for a passivated block or channel by setting the DEPASS input. For this, the RED_DEPA function (FC 451) is called internally.

## Redundancy

Redundancy of the modules is monitored in a higher-level RED_STATUS block (FB 453).

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

The OMODE_xx outputs are updated when the "Startup" bit is set. ALARM_8P will be initialized.

## Time response

Not available

## Message response

OR_HA16C uses ALARM_8P for reporting. The message function can be disabled by setting EN_MSG = FALSE.

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

Note: Online help and the manual "PCS 7 FACEPLATES" are only available if the "PCS 7 FACEPLATES" software package is installed.

## Additional information

You will find more information in:

Message texts and associated values of OR_HA16C (Page 253)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

### 3.34.2 I/Os of OR_M_8C / OR_M_16C / OR_M_32C / OR_HA16C

The OR_M_8C and OR_M_16C / OR_HA16C / OR_M_32C block I/Os are identical, with the exception of the number of MODE1_xx and MODE2_xx and OMODE_xx.

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **ACTIV_H** | 1 = module with more significant address is active | BOOL | 0 |
| **ACTIV_L** | 1 = module with less significant address is active | BOOL | 0 |
| **CHAN_NUM** | Number of channels -1 | INT | 7 (OR_M_8C)<br>15 (OR_M_16C)<br>31 (OR_M_32C)<br>15 (OR_HA16C) |
| CH_INF_H | Status of the channels on the 2nd module; info on a channel basis<br>0 = passivated, 1 = in service | DWORD | 0 |
| CH_INF_L | Status of the channels on the 1st module; info on a channel basis<br>0 = passivated, 1 = in service | DWORD | 0 |
| DEPASS | 1 = depassivation | BOOL | 0 |
| DEPASS_EN | 1 = enable depassivation | BOOL | 1 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number (x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | DWORD | 0 |
| **MOD_STAT** | Module status word from RED_STATUS | WORD | 0 |
| MODE1_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31) on the primary module<br>HART variable mode x (x = 1 - 8) on the primary module | DWORD | 0 |
| MODE2_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31) on the redundant module<br>HART variable mode x (x = 1 - 8) on the redundant module | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACKF1** | 1 = error rack 1 | BOOL | 0 |
| **RACKF2** | 1 = error rack 2 | BOOL | 0 |
| **RED_STAT** | Value returned by the RED_STATUS block | INT | 0 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_ALM | Channel failure Redundant pair | DWORD | 0 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| CH_WRN | Channel redundancy loss | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 07 / 00 - 15 / 00 - 31 / 00 - 15)<br>Bit 0= Release for maintenance<br>Bit 1-31 = Reserve | DWORD | 0 |
| EXT_INFO | Supplemental info of RED_OUT | INT | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| MSG_ACKx | Message acknowledgment ALARM_8P_x<br>(x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | WORD | 0 |
| MSG_STATx | Message error information ALARM_8P_x<br>(x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31)<br>mode HART variable x (x = 1 - 8) | DWORD | 0 |
| QDISCREP | 1 = At least one channel pair does not match | BOOL | 0 |
| QERR | 1 = program runtime error<br>(cannot determine module status) | BOOL | 1 |
| QMODF1 | 1 = error module 1 | BOOL | 0 |
| QMODF2 | 1 = error module 2 | BOOL | 0 |
| QPASS | 1 = at least one module is passivated | BOOL | 0 |
| RETURN_VAL | Error information from RED_OUT | INT | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| DXCHG1_xx | Bidirectional data exchange channel 1<br>xx = 00 to 07 (OR_M8C and OR_HA16C<br>xx = 00 to 15 (OR_M16C)<br>xx = 00 to 31 (OR_32C) | DWORD | 0 |
| DXCHG2_xx | Bidirectional data exchange channel 2<br>xx = 00 to 07 (OR_M8C and OR_HA16C<br>xx = 00 to 15 (OR_M16C)<br>xx = 00 to 31 (OR_32C) | DWORD | 0 |
| MOD_INF1 | Module parameters module 1 | STRUCT | |
| MOD_INF2 | Module parameters module 2 | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Maintenance Status of MS (Page 403)

Message texts and associated values of OR_M_8C (Page 278)

Message texts and associated values of OR_M_16C (Page 260)

Message texts and associated values of OR_HA16C (Page 253)

Message texts and associated values of OR_M_32C (Page 267)

General Information About Block Description (Page 9)

### 3.34.3 Message texts and associated values of OR_HA16C

#### Assignment of message text and message classes

You will find more information in Message classes (Page 401)

| Message block ALARM_8P | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Failure of module redundancy pair | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Module redundancy loss | F |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Cannot determine module status | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>At least one channel pair does not match | M |
| EV_ID2 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 00 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 01 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 02 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 03 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 04 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 05 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 06 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 07 | S |
| EV_ID3 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure HART variable 01 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure HART variable 02 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure HART variable 03 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure HART variable 04 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure HART variable 05 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure HART variable 06 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure HART variable 07 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure HART variable 08 | S |

| Message block ALARM_8P | Message no. | Default message text | Message class |
| --- | --- | --- | --- |
| EV_ID4 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 00 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 01 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 02 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 03 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 04 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 05 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 06 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 07 | F |
| EV_ID5 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss HART variable 01 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss HART variable 02 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss HART variable 03 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss HART variable 04 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss HART variable 05 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss HART variable 06 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss HART variable 07 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss HART variable 08 | F |

## Assignment of Associated Values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID1 ... 5 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |

Redundancy loss causes the geographic address of the failed module to be entered dynamically.

If both modules fail, the message text will always contain the geographic address of the primary module.

# 3.35 OR_M_16C: OR value status of 2 redundant signal modules, max. 16 channels, channel granular

## 3.35.1 Description of OR_M_16

### Object name (type + number)

FB 84

- OR_M_16C block I/Os (Page 257)

The OR_M_16C block corresponds to OR_M_8C (Page 272), but with 16, rather than 8 channels.

### Additional information

For further information, refer to the following sections:

Message texts and associated values of OR_M_16C (Page 260)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Maintenance release

## 3.35.2 I/Os of OR_M_8C / OR_M_16C / OR_M_32C / OR_HA16C

The OR_M_8C and OR_M_16C / OR_HA16C / OR_M_32C block I/Os are identical, with the exception of the number of MODE1_xx and MODE2_xx and OMODE_xx.

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **ACTIV_H** | 1 = module with more significant address is active | BOOL | 0 |
| **ACTIV_L** | 1 = module with less significant address is active | BOOL | 0 |
| **CHAN_NUM** | Number of channels -1 | INT | 7 (OR_M_8C)<br>15 (OR_M_16C)<br>31 (OR_M_32C)<br>15 (OR_HA16C) |
| CH_INF_H | Status of the channels on the 2nd module; info on a channel basis<br>0 = passivated, 1 = in service | DWORD | 0 |
| CH_INF_L | Status of the channels on the 1st module; info on a channel basis<br>0 = passivated, 1 = in service | DWORD | 0 |
| DEPASS | 1 = depassivation | BOOL | 0 |
| DEPASS_EN | 1 = enable depassivation | BOOL | 1 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number (x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | DWORD | 0 |
| **MOD_STAT** | Module status word from RED_STATUS | WORD | 0 |
| MODE1_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31) on the primary module<br>HART variable mode x (x = 1 - 8) on the primary module | DWORD | 0 |
| MODE2_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31) on the redundant module<br>HART variable mode x (x = 1 - 8) on the redundant module | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACKF1** | 1 = error rack 1 | BOOL | 0 |
| **RACKF2** | 1 = error rack 2 | BOOL | 0 |
| **RED_STAT** | Value returned by the RED_STATUS block | INT | 0 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_ALM | Channel failure Redundant pair | DWORD | 0 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| CH_WRN | Channel redundancy loss | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 07 / 00 - 15 / 00 - 31 / 00 - 15)<br>Bit 0= Release for maintenance<br>Bit 1-31 = Reserve | DWORD | 0 |
| EXT_INFO | Supplemental info of RED_OUT | INT | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| MSG_ACKx | Message acknowledgment ALARM_8P_x<br>(x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | WORD | 0 |
| MSG_STATx | Message error information ALARM_8P_x<br>(x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31)<br>mode HART variable x (x = 1 - 8) | DWORD | 0 |
| QDISCREP | 1 = At least one channel pair does not match | BOOL | 0 |
| QERR | 1 = program runtime error<br>(cannot determine module status) | BOOL | 1 |
| QMODF1 | 1 = error module 1 | BOOL | 0 |
| QMODF2 | 1 = error module 2 | BOOL | 0 |
| QPASS | 1 = at least one module is passivated | BOOL | 0 |
| RETURN_VAL | Error information from RED_OUT | INT | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| DXCHG1_xx | Bidirectional data exchange channel 1<br>xx = 00 to 07 (OR_M8C and OR_HA16C<br>xx = 00 to 15 (OR_M16C)<br>xx = 00 to 31 (OR_32C) | DWORD | 0 |
| DXCHG2_xx | Bidirectional data exchange channel 2<br>xx = 00 to 07 (OR_M8C and OR_HA16C<br>xx = 00 to 15 (OR_M16C)<br>xx = 00 to 31 (OR_32C) | DWORD | 0 |
| MOD_INF1 | Module parameters module 1 | STRUCT | |
| MOD_INF2 | Module parameters module 2 | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Maintenance Status of MS (Page 403)

Message texts and associated values of OR_M_8C (Page 278)

Message texts and associated values of OR_M_16C (Page 260)

Message texts and associated values of OR_HA16C (Page 253)

Message texts and associated values of OR_M_32C (Page 267)

General Information About Block Description (Page 9)

### 3.35.3 Message texts and associated values of OR_M_16C

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Failure of module redundancy pair | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Module redundancy loss | F |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Cannot determine module status | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>At least one channel pair does not match | M |
| EV_ID2 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 00 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 01 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 02 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 03 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 04 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 05 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 06 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 07 | S |
| EV_ID3 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 08 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 09 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 10 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 11 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 12 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 13 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 14 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 15 | S |

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID4 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 00 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 01 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 02 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 03 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 04 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 05 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 06 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 07 | F |
| EV_ID5 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 08 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 09 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 10 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 11 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 12 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 13 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 14 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 15 | F |

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID1 ... 5 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |

Redundancy loss causes the geographic address of the failed module to be entered dynamically.

If both modules fail, the message text will always contain the geographic address of the primary module.

## See also

Message Classes (Page 401)

# 3.36 OR_M_32C: OR value status of 2 redundant signal modules, max. 32 channels, channel granular

## 3.36.1 Description of OR_M_32C

### Object name (type + number)

FB85

- OR_M_32C block I/Os (Page 264)

Block OR_M_32C corresponds to the OR_M_8C (Page 272) block, but with 32 channels instead of 8.

### Additional information

You will find more information in:

Message texts and associated values of OR_M_32C (Page 267)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

### 3.36.2 I/Os of OR_M_8C / OR_M_16C / OR_M_32C / OR_HA16C

The OR_M_8C and OR_M_16C / OR_HA16C / OR_M_32C block I/Os are identical, with the exception of the number of MODE1_xx and MODE2_xx and OMODE_xx.

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **ACTIV_H** | 1 = module with more significant address is active | BOOL | 0 |
| **ACTIV_L** | 1 = module with less significant address is active | BOOL | 0 |
| **CHAN_NUM** | Number of channels -1 | INT | 7 (OR_M_8C)<br>15 (OR_M_16C)<br>31 (OR_M_32C)<br>15 (OR_HA16C) |
| CH_INF_H | Status of the channels on the 2nd module; info on a channel basis<br>0 = passivated, 1 = in service | DWORD | 0 |
| CH_INF_L | Status of the channels on the 1st module; info on a channel basis<br>0 = passivated, 1 = in service | DWORD | 0 |
| DEPASS | 1 = depassivation | BOOL | 0 |
| DEPASS_EN | 1 = enable depassivation | BOOL | 1 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number (x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | DWORD | 0 |
| **MOD_STAT** | Module status word from RED_STATUS | WORD | 0 |
| MODE1_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31) on the primary module<br>HART variable mode x (x = 1 - 8) on the primary module | DWORD | 0 |
| MODE2_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31) on the redundant module<br>HART variable mode x (x = 1 - 8) on the redundant module | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACKF1** | 1 = error rack 1 | BOOL | 0 |
| **RACKF2** | 1 = error rack 2 | BOOL | 0 |
| **RED_STAT** | Value returned by the RED_STATUS block | INT | 0 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_ALM | Channel failure Redundant pair | DWORD | 0 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| CH_WRN | Channel redundancy loss | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 07 / 00 - 15 / 00 - 31 / 00 - 15)<br>Bit 0= Release for maintenance<br>Bit 1-31 = Reserve | DWORD | 0 |
| EXT_INFO | Supplemental info of RED_OUT | INT | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| MSG_ACKx | Message acknowledgment ALARM_8P_x<br>(x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | WORD | 0 |
| MSG_STATx | Message error information ALARM_8P_x<br>(x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31)<br>mode HART variable x (x = 1 - 8) | DWORD | 0 |
| QDISCREP | 1 = At least one channel pair does not match | BOOL | 0 |
| QERR | 1 = program runtime error<br>(cannot determine module status) | BOOL | 1 |
| QMODF1 | 1 = error module 1 | BOOL | 0 |
| QMODF2 | 1 = error module 2 | BOOL | 0 |
| QPASS | 1 = at least one module is passivated | BOOL | 0 |
| RETURN_VAL | Error information from RED_OUT | INT | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| DXCHG1_xx | Bidirectional data exchange channel 1<br>xx = 00 to 07 (OR_M8C and OR_HA16C<br>xx = 00 to 15 (OR_M16C)<br>xx = 00 to 31 (OR_32C) | DWORD | 0 |
| DXCHG2_xx | Bidirectional data exchange channel 2<br>xx = 00 to 07 (OR_M8C and OR_HA16C<br>xx = 00 to 15 (OR_M16C)<br>xx = 00 to 31 (OR_32C) | DWORD | 0 |
| MOD_INF1 | Module parameters module 1 | STRUCT | |
| MOD_INF2 | Module parameters module 2 | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Maintenance Status of MS (Page 403)

Message texts and associated values of OR_M_8C (Page 278)

Message texts and associated values of OR_M_16C (Page 260)

Message texts and associated values of OR_HA16C (Page 253)

Message texts and associated values of OR_M_32C (Page 267)

General Information About Block Description (Page 9)

## 3.36.3 Message texts and associated values of OR_M_32C

### Assignment of message text and message class

| Message block ALARM_8P | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Failure of module redundancy pair | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Module redundancy loss | F |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Cannot determine module status | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>At least one channel pair does not match | M |
| EV_ID2 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 00 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 01 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 02 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 03 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 04 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 05 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 06 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 07 | S |
| EV_ID3 | 1 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 08 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 09 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 10 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 11 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 12 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 13 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 14 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@:<br>Redundancy pair failure channel 15 | S |

| Message block ALARM_8P | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID4 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 16 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 17 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 18 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 19 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 20 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 21 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 22 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 23 | S |
| EV_ID5 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 24 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 25 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 26 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 27 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 28 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 29 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 30 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 31 | S |

| Message block ALARM_8P | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID6 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 00 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 01 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 02 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 03 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 04 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 05 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 06 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 07 | F |
| EV_ID7 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 08 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 09 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 10 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 11 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 12 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 13 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 14 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 15 | F |

| Message block ALARM_8P | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID8 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 16 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 17 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 18 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 19 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 20 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 21 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 22 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 23 | F |
| EV_ID9 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 24 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 25 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 26 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 27 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 28 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 29 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 30 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 31 | F |

## Assignment of Associated Values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID1 ... 9 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |

Redundancy loss causes the geographic address of the failed module to be entered dynamically.

If both modules fail, the message text will always contain the geographic address of the primary module.

## See also

Message Classes (Page 401)

# 3.37 OR_M_8C: OR value status of 2 redundant signal modules, max. 8 channels, channel granular

## 3.37.1 Description of OR_M_8C

### Object name (type + number)

FB 83

- OR_M_8C block I/Os (Page 275)

### Area of application

The OR_M_8C block generates a channel-granular value status from two redundant signal modules.

### Calling OBs

The block must be installed in OB 100 and in the fastest OB upstream of the CH_x block that is interconnected with OR_M_8C.

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- The OR_M_8C block is installed upstream in the OBs of the CH_x channel blocks that are interconnected with it.
- MODE1_x inputs are interconnected with the OMODE_xx outputs of the MOD_x block in the primary module.
- MODE2_x inputs are interconnected with the OMODE_xx outputs of the MOD_x block in the redundant module.
- The MOD_INF1 input structure is interconnected with the MOD_INF output structure of the MOD_x block in the primary module.
- The MOD_INF2 input structure is interconnected with the MOD_INF output structure of the MOD_x block in the redundant module.
- The ACTIV_H and ACTIV_L inputs are interconnected with the outputs of the same name of the RED_STATUS block in the redundant module.
- The OMODE_xx outputs are interconnected with the relevant CH_x channel blocks.
- The OUT structure CPU_DIAG of the OB_BEGIN block is interconnected with the IN_OUT structures of the same name of the OR_M_8C block.
- The RACKF1 and RACKF2 inputs are interconnected with the QRACKF1 and QRACKF2 outputs of MOD_xx.

- The CH_INF_H and CH_INF_L inputs are interconnected with the outputs of the same name of the RED_STATUS block.
- The output parameter of DXCHG_xx is interconnected with the following channel block at the DataXchg parameter.
- The output parameter of O_MS is interconnected with the following channel block at the MS parameter.

## Function and method of operation

The value statuses of a signal module and of a redundant signal module are ORed by the OR_M_8C block. Signal modules or signal channels set to passive mode by the system are regarded as invalid. For redundant digital input modules, when a signal discrepancy occurs, no module or channel is set to passive after the discrepancy time has elapsed. The module or channel whose signal does not change is then set to passive. The "At least one channel pair does not match", "Redundancy loss" and "Failure of the redundant I/O" events are reported by ALARM_8P. The message function can be disabled.

Note: When module signals are processed in different OB 3x, in very few cases, a faulty signal value may be processed in one cycle by one of the channel blocks configured to process the module signals following a process-control malfunction affecting the module. This can be avoided by processing all channel blocks of a module in the OB 3x that also processes the process image partition to which this module is assigned.

For OR_M_Cxx blocks:

With channel-granular redundancy, "Loss of redundancy channel x" is reported if one channel fails. If both channels fail, "Redundancy pair failure channel x" is reported. The message function can be disabled.

For all OR blocks:

Depassivation can be triggered for a passivated block or channel by setting the DEPASS input. This is done by calling the RED_DEPA FC451 function internally.

## Redundancy

Redundancy of the modules in an H system is monitored in the higher-level RED_STATUS block.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

The OMODE_xx (Page 396) outputs are updated when the "Startup" bit is set.

ALARM_8P will be initialized.

## Message response

OR_M_8C uses ALARM_8P for reporting. The message function can be disabled by setting EN_MSG = FALSE.

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of OR_M_8C (Page 278)

Maintenance status of MS (Page 403)

You will find more information on this in PCS 7 Advanced Process Library > Basics of APL > General functions of the blocks > Operating, monitoring and reporting > Release for maintenance

## 3.37.2 I/Os of OR_M_8C / OR_M_16C / OR_M_32C / OR_HA16C

The OR_M_8C and OR_M_16C / OR_HA16C / OR_M_32C block I/Os are identical, with the exception of the number of MODE1_xx and MODE2_xx and OMODE_xx.

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **ACTIV_H** | 1 = module with more significant address is active | BOOL | 0 |
| **ACTIV_L** | 1 = module with less significant address is active | BOOL | 0 |
| **CHAN_NUM** | Number of channels -1 | INT | 7 (OR_M_8C)<br>15 (OR_M_16C)<br>31 (OR_M_32C)<br>15 (OR_HA16C) |
| CH_INF_H | Status of the channels on the 2nd module; info on a channel basis<br>0 = passivated, 1 = in service | DWORD | 0 |
| CH_INF_L | Status of the channels on the 1st module; info on a channel basis<br>0 = passivated, 1 = in service | DWORD | 0 |
| DEPASS | 1 = depassivation | BOOL | 0 |
| DEPASS_EN | 1 = enable depassivation | BOOL | 1 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number (x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | DWORD | 0 |
| **MOD_STAT** | Module status word from RED_STATUS | WORD | 0 |
| MODE1_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31) on the primary module<br>HART variable mode x (x = 1 - 8) on the primary module | DWORD | 0 |
| MODE2_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31) on the redundant module<br>HART variable mode x (x = 1 - 8) on the redundant module | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACKF1** | 1 = error rack 1 | BOOL | 0 |
| **RACKF2** | 1 = error rack 2 | BOOL | 0 |
| **RED_STAT** | Value returned by the RED_STATUS block | INT | 0 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_ALM | Channel failure Redundant pair | DWORD | 0 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| CH_WRN | Channel redundancy loss | DWORD | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 07 / 00 - 15 / 00 - 31 / 00 - 15)<br>Bit 0= Release for maintenance<br>Bit 1-31 = Reserve | DWORD | 0 |
| EXT_INFO | Supplemental info of RED_OUT | INT | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| MSG_ACKx | Message acknowledgment ALARM_8P_x<br>(x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | WORD | 0 |
| MSG_STATx | Message error information ALARM_8P_x<br>(x = 1 – 3 / 1 – 5 / 1 - 9 / 1 - 5) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Channel mode (xx = 00 – 07 / 00 – 15 / 00 - 31)<br>mode HART variable x (x = 1 - 8) | DWORD | 0 |
| QDISCREP | 1 = At least one channel pair does not match | BOOL | 0 |
| QERR | 1 = program runtime error<br>(cannot determine module status) | BOOL | 1 |
| QMODF1 | 1 = error module 1 | BOOL | 0 |
| QMODF2 | 1 = error module 2 | BOOL | 0 |
| QPASS | 1 = at least one module is passivated | BOOL | 0 |
| RETURN_VAL | Error information from RED_OUT | INT | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| DXCHG1_xx | Bidirectional data exchange channel 1<br>xx = 00 to 07 (OR_M8C and OR_HA16C<br>xx = 00 to 15 (OR_M16C)<br>xx = 00 to 31 (OR_32C) | DWORD | 0 |
| DXCHG2_xx | Bidirectional data exchange channel 2<br>xx = 00 to 07 (OR_M8C and OR_HA16C<br>xx = 00 to 15 (OR_M16C)<br>xx = 00 to 31 (OR_32C) | DWORD | 0 |
| MOD_INF1 | Module parameters module 1 | STRUCT | |
| MOD_INF2 | Module parameters module 2 | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Maintenance Status of MS (Page 403)

Message texts and associated values of OR_M_8C (Page 278)

Message texts and associated values of OR_M_16C (Page 260)

Message texts and associated values of OR_HA16C (Page 253)

Message texts and associated values of OR_M_32C (Page 267)

General Information About Block Description (Page 9)

### 3.37.3 Message texts and associated values of OR_M_8C

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | Module @1%d@/@2%d@/@3%d@: Failure of module redundancy pair | S |
| | 2 | Module @1%d@/@2%d@/@3%d@: Module redundancy loss | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Cannot determine module status | S |
| | 4 | Module @1%d@/@2%d@/@3%d@: At least one channel pair does not match | M |
| EV_ID2 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 00 | S |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 01 | S |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 02 | S |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 03 | S |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 04 | S |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 05 | S |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 06 | S |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy pair failure channel 07 | S |
| EV_ID3 | 1 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 00 | F |
| | 2 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 01 | F |
| | 3 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 02 | F |
| | 4 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 03 | F |
| | 5 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 04 | F |
| | 6 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 05 | F |
| | 7 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 06 | F |
| | 8 | Module @1%d@/@2%d@/@3%d@: Redundancy loss channel 07 | F |

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameters | Meaning |
|---|---|---|---|
| EV_ID1 / 2 / 3 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | SLOT_NO | Slot number (byte) |

Redundancy loss causes the geographic address of the failed module to be entered dynamically.

If both modules fail, the message text will always contain the geographic address of the primary module.

## See also

Message Classes (Page 401)

# 3.38 PADP_L0x: Monitoring DP/PA slaves

## 3.38.1 Description of PADP_L00

### Object name (type + number)

FB 109

- PADP_L00 block I/Os (Page 284)

### Area of application

Block PADP_L00 monitors DP/PA field devices operating as DPV0 or DPV1 slaves downstream of a DP/PA or Y-Link that is operated as a DPV0 slave. The PA field devices must conform to the PROFIBUS V3.0 profile. Individual blocks must be available for the diagnostic and signal processing functions of DP field devices. H systems support only the PA field devices at an active DP/PA-Link.

### Calling OBs

The block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- Block PADP_L00 is integrated in the run sequence downstream of block DPAY_V0.
- Parameters are assigned to the MODE_xx input (mode of slot xx of a field device).
- The PADP_ADR input (DP/PA slave address downstream of a DP/PA or Y-Link) is configured.
- The input DPA_M is interconnected with the output DPA_M_xx of the block DPAY_V0.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the DPAY_V0 block are interconnected with the IN_OUT structures of the same name of PADP_L00.
- The QMODF and PA_DIAG outputs are interconnected with the MOD_PAL0 block.

**Note**

The CFC function "**Generate module drivers**" can only be used if the PA field device belongs to slave family 12.

## Function and method of operation

Block PADP_L00 analyzes all events affecting a DP or PA field device and its slots acyclically. It generates the relevant DP_MODE or PA_MODE, depending on the slot, and the value status for the DP or PA signal processing blocks. The permitted PA_MODE (Page 397) is already defined for the PA signal processing blocks. while DP_MODE has to be defined individually for the blocks of the DP field devices. ALARM_8P is used to report these events. The message function can be disabled.

The higher-level DPAY_V0 block enables the block to run. The diagnostic is entered in the start information (CPU_DIAG) of the OB_BEGIN block.

The data is already evaluated by the link block (DPAY_V0) if diagnostics are required. The diagnostic information concerning the PA field device is stored in the DPA_M structure. The structure consists of two DWORD (S_01 for modules 1 to 16 and S_02 for modules 17 to 32) and one BOOL (S_ERR = DP/PA field device faulty) variables. Two bits of the DWORD are assigned to each slot of the DP/PA field device, whereby bit 0 and bit 1 belong to slot 1 of the DP/PA field device, etc. Slots 1 to 7 are evaluated. The bit states are defined as follows:

| Status Bit 0 | Status Bit 1 | Meaning |
|---|---|---|
| 0 | 0 | Module x OK (valid user data) |
| 0 | 1 | Module x error (invalid user data) |
| 1 | 0 | Wrong module x (invalid user data) |
| 1 | 1 | No module x (invalid user data) |

An input (MODE_xx) exists for each slot (module) of the DP/PA field device that is used to read in configuration settings made for the PA field device slots (module) in HW Config.

At DP field devices the user must enter his code manually at the MODE input.

The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current slot value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx. The following events are evaluated by block DPAY_V0 and lead to the value status “Invalid value” due to a higher-level error (OMODE_xx = 16#40xxxxxx):

| | |
|---|---|
| • Rack failure (OB 86) | (output parameter QRACKF = TRUE) |
| • Diagnostic interrupt affecting an entire field device (OB 82) | (output parameter QMODF = TRUE if DPA_M.S_ERR = TRUE) |
| • Diagnostic interrupt slot xx affecting a field device (OB 82): | (output parameter OMODE_xx = Module-(slot-)error-specific DPA_M) |

In the event of a diagnostic interrupt, the block reports field-device-specifically to WinCC by means of ALARM_8P. We distinguish between the field device and its slots; each slot is assigned a message ID.

The “Device failure” message can be disabled with EM_MSG_D = FALSE.

## Redundancy

Higher-level block DPAY_V0 evaluates the redundancy of DP master systems operating in an H system.

## MODE Setting for PA Profiles

You can find additional information in "PA_MODE Settings (Page 397)".

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

## OMODE Structure

You can find additional information in "OMODE (Page 396)".

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

After a restart or an initial start the system verifies that the PA field device is available under its logical base address. A restart (OB100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

The block uses ALARM_8P to report field device errors and generates the following messages in the OBs listed below:

| OB no. | Start Event | Message |
|---|---|---|
| OB 1 | Cyclic processing | Repeat the update of ALARM_8P outputs/messages, if necessary |
| OB 82 | Diagnostic interrupt | Device error incoming/outgoing<br>Device module xx error incoming/outgoing<br>Device module xx incorrect incoming/outgoing<br>Device module xx missing incoming/outgoing |
| OB 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

The block has no faceplate.

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

## Additional information

For further information, refer to the following sections:

Message texts and associated values for PADP_L00 (Page 286)

## 3.38.2 I/Os of PADP_L00/PADP_L01/PADP_L02

### I/Os

The I/Os of the PADP_L00, PADP_L01 and PADP_L02 blocks are identical save for the number of MODE_xx and OMODE_xx. The number of monitored slots determines the number of corresponding I/O parameters.

The default block view in the CFC is identified in the "I/O" column:
I/O name in **bold** = I/O is visible, standard I/O name = I/O is not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| **DPA_M** | DP/PA field device diagnostic information | STRUCT | |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EN_MSG_D | 1 = enable "Device failure" message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| MODE_xx | Module mode (xx = 00 - 06 / 00 - 15 / 00 - 31) | WORD | 0 |
| **PADP_ADR** | DP/PA field device address | BYTE | 0 |

### Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSGSTAT | Message error information | WORD | 0 |
| OMODE_xx | Module mode (xx = 00 - 06 / 00 - 15 / 00 - 31) | DWORD | 0 |
| **PA_DIAG** | PA field device diagnostic information | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| **QMODF** | 1 = module removed/defective | BOOL | 0 |
| **QRACKF** | 1 = rack/station error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | CPU diagnostics | STRUCT | |
| RAC_DIAG | 1 = DPV1 mode | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values for PADP_L00 (Page 286)

Message texts and associated values for PADP_L01 (Page 291)

Message texts and associated values for PADP_L02 (Page 297)

### 3.38.3 Message texts and associated values of PADP_L00

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID | 1 | QMODF | Device @1%d@/ @2%d@/@3%d@: Failure | S |
| | 2 | - | Device @1%d@/ @2%d@/@3%d@: Module 00 @4W%t#PADP_L00_TXT@ | S |
| | ... | | ... | |
| | 8 | - | Device @1%d@/ @2%d@/@3%d@: Module 06 @10W%t#PADP_L00_TXT@ | S |

You will find the message texts and their text numbers in the

"Text library for PADP_L00 (Page 416)" section.

#### Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 10 | - | Text number (message 2 - 8) from PADP_L00_TXT |

If the PA field device is connected downstream from an inactive DP/PA-Link V0and SUBN1_ID = 16#FF, the associated variable is substituted by SUBN2_ID.

#### See also

Message Classes (Page 401)

## 3.38.4 Description of PADP_L01

### Object name (type + number)

FB 110

- PADP_L01 block I/Os (Page 284)

### Area of application

The PADP_L01 monitors DP/PA field devices that are used as DPV0 or DPV1 slaves, downstream of a DP/PA or Y link that is used as a DPV0 slave. The PA field devices must conform to the PROFIBUS V3.0 profile. There must be individual blocks available for the diagnostics and signal processing for DP field devices. H systems support only the PA field devices at an active DP/PA-Link.

### Calling OBs

The block must be installed in the run sequence in the following OBs (this is done automatically in the CC):

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The PADP_L01 block is integrated in the run sequence downstream of block DPAY_V0.
- Parameters are assigned to the MODE_xx input (mode of slot xx of a field device).
- The PADP_ADR input (DP/PA slave address downstream of the DP/PA link or Y link) is configured.
- The DPA_M input is interconnected with the DPA_M_xx output of the DPAY_V0 block.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the DPAY_V0 block are interconnected with the IN_OUT structures of the same name of PADP_L01.
- The outputs QMODF and PA_DIAG are interconnected with the MOD_PAL0 block.

---

**Note**

The CFC function "Generate module drivers" can only be used if the PA field device belongs to slave family 12.

---

## Function and method of operation

Block PADP_L01 analyzes all events affecting a DP or PA field device and its slots acyclically. It generates the relevant DP_MODE or PA_MODE, depending on the slot, and the value status for the DP or PA signal processing blocks. The permitted PA_MODE (Page 397) is already defined for the PA signal processing blocks. The DP_MODE must be defined individually for the DP field device blocks. ALARM_8P is used to report these events. The message function can be disabled.

The higher-level DPAY_V0 block enables the block to run. The diagnostic event is entered in the start information (CPU_DIAG) of the OB_BEGIN block.

The data is already evaluated by the link block (DPAY_V0) if diagnostics are required. The diagnostic information concerning the PA field device is stored in the DPA_M structure. The structure consists of 2 DWORD (S_01 for module 1 to 16 and S_02 for module 17 to 32), and 1 BOOL (S_ERR = DP/PA field device faulty) variables. Two bits of the DWORD are assigned to each slot of the DP/PA field device, whereby bit 0 and bit 1 belong to slot 1 of the DP/PA field device, etc. Slots 1 to 16 are evaluated. They are defined as follows:

| Status Bit 0 | Status Bit 1 | Meaning |
|---|---|---|
| 0 | 0 | Module x OK (valid user data) |
| 0 | 1 | Module x error (invalid user data) |
| 1 | 0 | Wrong module x (invalid user data) |
| 1 | 1 | No module x (invalid user data) |

There is an input (MODE_xx (Page 387)) for each slot (module) on the DP/PA field device that is used to read in configuration settings made for the PA field device slots (module) in HW Config.

For DP field devices, the user must do his own encoding at the MODE input.

The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current slot value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx. The following events are evaluated by block DPAY_V0 and lead to the value status “Invalid value” due to a higher-level error (OMODE_xx = 16#40xxxxxx):

| | |
|---|---|
| • Rack failure (OB 86) | (Output parameter QRACKF = TRUE) |
| • Diagnostic interrupt that affects entire field device (OB 82) | (Output parameter QMODF = TRUE, if DPA_M.S_ERR = TRUE) |
| • Diagnostic interrupt slot xx that affects a field device (OB 82): | (Output parameter OMODE_xx = Module (slot) error-specific DPA_M) |

The block reports a diagnostic interrupt to the OS for a specific field device using ALARM_8P. We distinguish between the field device and its slots; each slot is assigned a message ID.

The “Device failure” message can be disabled with EM_MSG_D = FALSE.

## Redundancy

The higher-level block DPAY_V0 evaluates the redundancy of the DP master systems used in an H system.

## MODE setting for PA profiles

You will find further information in "PA_MODE settings (Page 397)".

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

## OMODE structure

You will find further information in the "OMODE (Page 396)" section.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

After a restart/initial startup, the system checks that the PA field device is available at its logical base address. A restart (OB100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

The block signals field device errors using ALARM_8P, and generates the following messages in the OBs listed below:

| OB no. | Start Event | Message |
|---|---|---|
| OB 1 | Cyclic processing | Repeat the update of ALARM_8P outputs/messages, if necessary |
| OB 82 | Diagnostic interrupt | Device error incoming/outgoing<br>Device module xx error incoming/outgoing<br>Device module xx incorrect incoming/outgoing<br>Device module xx missing incoming/outgoing |
| OB 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

The block has no faceplate.

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

## Additional information

For further information, refer to the following sections:

Message texts and associated values for PADP_L01 (Page 291)

## 3.38.5 Message texts and associated values of PADP_L01

### Assignment of message text and message class

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 | 1 | QMODF | Device @1%d@/ @2%d@/@3%d@: Failure | S |
| | 2 | - | Device @1%d@/ @2%d@/@3%d@: Module 07 @4W%t#PADP_L01_TXT@ | S |
| | 3 | - | Device @1%d@/ @2%d@/@3%d@: Module 15 @5W%t#PADP_L01_TXT@ | S |
| EV_ID2 | 1 | - | Device @1%d@/ @2%d@/@3%d@: Module 00 @4W%t#PADP_L01_TXT@ | S |
| | ... | | ... | |
| | 7 | - | Device @1%d@/ @2%d@/@3%d@: Module 06 @10W%t#PADP_L01_TXT@ | S |
| EV_ID3 | 1 | - | Device @1%d@/ @2%d@/@3%d@: Module 08 @4W%t#PADP_L01_TXT@ | S |
| | ... | | ... | |
| | 7 | - | Device @1%d@/ @2%d@/@3%d@: Module 14 @10W%t#PADP_L01_TXT@ | S |

You will find the message texts and their text numbers in the "Text library for PADP_L01 (Page 416)" section.

## Assignment of associated values

| Message block ALARM_8P | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID1 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 5 | - | Text number (message 2 - 3) from PADP_L01_TXT |
| EV_ID2 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 10 | - | Text number (message 1 - 7) from PADP_L01_TXT |
| EV_ID3 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 10 | - | Text number (message 1 - 7) from PADP_L01_TXT |

If the PA field device is connected downstream from an inactive DP/PA-Link V0and SUBN1_ID = 16#FF, the associated variable is substituted by SUBN2_ID.

## See also

Message Classes (Page 401)

## 3.38.6 Description of PADP_L02

### Object name (type + number)

FB 111

- PADP_L02 block I/Os (Page 284)

### Area of application

The PADP_L02 monitors DP/PA field devices that are used as DPV0 or DPV1 slaves, downstream of a DP/PA or Y link that is used as a DPV0 slave. The PA field devices must conform to the PROFIBUS V3.0 profile. There must be individual blocks available for the diagnostics and signal processing for DP field devices. H systems support only the PA field devices at an active DP/PA-Link.

### Calling OBs

The PADP_L02 block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 82 | Diagnostic interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The PADP_L02 block is integrated in the run sequence downstream of block DPAY_V0.
- Parameters are assigned to the MODE_xx input (mode of slot xx of a field device).
- The PADP_ADR input (DP/PA slave address downstream of the DP/PA link or Y link) is configured.
- The DPA_M input is interconnected with the DPA_M_xx output of the DPAY_V0 block.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the DPAY_V0 block are interconnected with the IN_OUT structures of the same name of PADP_L02.
- The QMODF and PA_DIAG outputs are interconnected with the MOD_PAL0 block.

## Function and method of operation

Block PADP_L02 analyzes all events affecting a DP or PA field device and its slots acyclically. It generates the relevant DP_MODE or PA_MODE, depending on the slot, and the value status for the DP or PA signal processing blocks. The permitted PA_MODE (Page 397) is already defined for the PA signal processing blocks. The DP_MODE must be defined individually for the DP field device blocks. ALARM_8P is used to report these events. The message function can be disabled.

The higher-level DPAY_V0 block enables the block to run. The event to be evaluated is entered in the start information (CPU_DIAG) of OB_BEGIN.

The data is already evaluated by the link block (DPAY_V0) if diagnostics are required. The diagnostic information concerning the PA field device is stored in the DPA_M structure. The structure consists of 2 DWORD (S_01 for module 1 to 16 and S_02 for module 17 to 32), and 1 BOOL (S_ERR = DP/PA field device faulty) variables. Two bits of the DWORD are assigned to each slot of the DP/PA field device, whereby bit 0 and bit 1 belong to slot 1 of the DP/PA field device, etc. Slots 1 to 32 are evaluated. They are defined as follows:

| Status Bit 0 | Status Bit 1 | Meaning |
|---|---|---|
| 0 | 0 | Module x OK (valid user data) |
| 0 | 1 | Module x error (invalid user data) |
| 1 | 0 | Wrong module x (invalid user data) |
| 1 | 1 | No module x (invalid user data) |

There is an input (MODE_xx) for each slot (module) on the DP/PA field device that is used to read in configuration settings made for the PA field device slots (module) in HW Config.

For DP field devices, the user must do his own encoding at the MODE input.

The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current slot value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx. The following events are evaluated by block DPAY_V0 and lead to the value status “Invalid value” due to a higher-level error (OMODE_xx = 16#40xxxxxx):

| | |
|---|---|
| • Rack failure (OB 86) | (Output parameter QRACKF = TRUE) |
| • Diagnostic interrupt that affects entire field device (OB 82) | (Output parameter QMODF = TRUE, if DPA_M.S_ERR = TRUE) |
| • Diagnostic interrupt slot xx that affects a field device (OB 82): | (Output parameter OMODE_xx = Module (slot) error-specific DPA_M) |

The block reports a diagnostic interrupt to WinCC for a specific field device using ALARM_8P. We distinguish between the field device and its slots; each slot is assigned a message ID.

The “Device failure” message can be disabled with EM_MSG_D = FALSE.

## Redundancy

The higher-level block DPAY_V0 evaluates the redundancy of the DP master systems used in an H system.

## MODE setting for PA profiles

You will find further information in "PA_MODE settings (Page 397)".

**Note**

If you change the parameter settings for the MODE_xx inputs at runtime, these changes will not be accepted at the outputs until the ACC_MODE is set to 1.

## OMODE Structure

You will find further information in the "OMODE (Page 396)" section.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

After a restart/initial startup, the system checks that the PA field device is available at its logical base address. A restart (OB100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

The block signals field device errors using ALARM_8P, and generates the following messages in the OBs listed below:

| OB no. | Start Event | Message |
|---|---|---|
| OB 1 | Cyclic processing | Repeat the update of ALARM_8P outputs/messages,<br>if necessary |
| OB 82 | Diagnostic interrupt | Device error incoming/outgoing<br>Device module xx error incoming/outgoing<br>Device module xx incorrect incoming/outgoing<br>Device module xx missing incoming/outgoing |
| OB 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

The block has no faceplate.

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

## Additional information

For further information, refer to the following sections:

Message texts and associated values for PADP_L02 (Page 297)

## 3.38.7 Message texts and associated values of PADP_L02

### Assignment of message text and message class

| Message block ALARM_8P | Message number | Block parameter | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 | 1 | QMODF | Device @1%d@/ @2%d@/@3%d@: Failure | S |
| | 2 | - | Device @1%d@/ @2%d@/@3%d@: Module 07 @4W%t#PADP_L02_TXT@ | S |
| | 3 | - | Device @1%d@/ @2%d@/@3%d@: Module 15 @5W%t#PADP_L02_TXT@ | S |
| | 4 | - | Device @1%d@/ @2%d@/@3%d@: Module 23 @6W%t#PADP_L02_TXT@ | S |
| | 5 | - | Device @1%d@/ @2%d@/@3%d@: Module 31 @7W%t#PADP_L02_TXT@ | S |
| EV_ID2 | 1 | - | Device @1%d@/ @2%d@/@3%d@: Module 00 @4W%t#PADP_L02_TXT@ | S |
| | ... | | ... | |
| | 7 | - | Device @1%d@/ @2%d@/@3%d@: Module 06 @10W%t#PADP_L02_TXT@ | S |
| EV_ID3 | 1 | - | Device @1%d@/ @2%d@/@3%d@: Module 08 @4W%t#PADP_L02_TXT@ | S |
| | ... | | ... | |
| | 7 | - | Device @1%d@/ @2%d@/@3%d@: Module 14 @10W%t#PADP_L02_TXT@ | S |
| EV_ID4 | 1 | - | Device @1%d@/ @2%d@/@3%d@: Module 16 @4W%t#PADP_L02_TXT@ | S |
| | ... | | ... | |
| | | - | Device @1%d@/ @2%d@/@3%d@: Module 22 @10W%t#PADP_L02_TXT@ | S |
| EV_ID5 | 1 | - | Device @1%d@/ @2%d@/@3%d@: Module 24 @4W%t#PADP_L02_TXT@ | S |
| | ... | | ... | |
| | 7 | - | Device @1%d@/ @2%d@/@3%d@: Module 30 @10W%t#PADP_L02_TXT@ | S |

You will find the message texts and their text numbers in "Text library for PADP_L02 (Page 416)".

### Assignment of associated values to the block parameters of PADP_L02

| Message block ALARM_8P | Associated value | Block parameter | Meaning |
|---|---|---|---|
| EV_ID1 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 7 | - | Text number (message 2 - 5) from PADP_L02_TXT |
| EV_ID2 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 10 | - | Text number (message 1 - 7) from PADP_L02_TXT |
| EV_ID3 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 10 | - | Text number (message 1 - 7) from PADP_L02_TXT |
| EV_ID4 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 10 | - | Text number (message 1 - 7) from PADP_L02_TXT |
| EV_ID5 | 1 | SUBN_ID | DP master system ID (byte) |
| | 2 | RACK_NO | Rack/station number (byte) |
| | 3 | PADP_ADR | DP/PA device address (byte) |
| | 4 - 10 | - | Text number (message 1 - 7) from PADP_L02_TXT |

If the PA field device is connected downstream from an inactive DP/PA-Link V0and SUBN1_ID = 16#FF, the associated variable is substituted by SUBN2_ID.

### See also

Message Classes (Page 401)

# 3.39 PADP_L10: Monitoring PA slaves downstream of DPV0 with up to 16 slots

## 3.39.1 Description of PADP_L10

### Object name (type + number)

FB 116

- PADP_L10 block I/Os (Page 305)

### Area of application

Block PADP_L10 monitors DPV0 PA field devices with a maximum of 32 slots, which are operated as DPV0 slaves on a DP master system, either directly or via a DP/PA coupler. The DP/PA coupler is connected downstream of a DPV1 DP/PA link. The PA field devices must conform to the PROFIBUS V3.0 profile. H systems support only the PA field devices at an active DP/PA-Link.

### Calling OBs

The block must be installed in the run sequence downstream from the OB_DIAG1 block in the following OBs (this is done automatically in the CFC):

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 55 | Status interrupt (only if a PA slave is required) |
| OB 56 | Update interrupt (only if a PA slave is required) |
| OB 57 | Vendor-specific interrupt (only if a PA slave is required) |
| OB 82 | Diagnostic interrupt |
| OB 83 | Remove/insert module interrupt (failure/return of a field device) |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- Block OB_DIAG1 is installed in the run sequence upstream of PADP_L10.
- The following are configured, depending on whether the PA field device is connected directly to a DP master system or downstream of a DP/PA link:
  - the diagnostic address of the PA field device or of the DADDR DP/PA link
  - the geographic address (SUBN1_ID, SUBN2_ID, RACK_NO of the PA address of the PA field device or the DP/PA link)
  - SLOT_NO = 0 or the geographic address of the PA field device in the DP/PA link (SLOT0_NO = Slot 0 no. for the PA field device, SLOTS_NO = number of slots of the PA field device)
  - SLOTS_NO = number of slots of the PA field device
  - the PA address of the PA field device (PADP_ADR)
  - MODE_xx (mode of slot xx of a PA field device)
- The CPU_DIAG and CPU_OB_5X OUT structures of OB_BEGIN and RAC_DIAG of OB_DIAG1 are interconnected with the IN_OUT structures of the same name of PADP_L10.
- The EN input is interconnected with the output of an AND block.
  whose inputs are interconnected with the outputs EN_SUBx (x = number of the DP master system) of the OB_BEGIN block, EN_Rxxx (xxx = rack/station number) of the SUBNET block, EN_Mx (x= number of the PA device) of the DPAY_V1 block, and EN_F of the OB_DIAG1 block.
- EN_DIAG is interconnected with output EN_DIAG of OB_DIAG1.
- Output QPERAF is interconnected with input PERAF of MOD_PAX0 or MOD_PAL0.
- Output QMODF is interconnected with input MODF of MOD_PAX0 or MOD_PAL0.
- Output PA_DIAG is interconnected with input PA_DIAG of MOD_PAX0 or MOD_PAL0.

---

**Note**

The CFC function "**Generate module drivers**" can only be used if the PA field device belongs to slave family 12.

---

## Description of the functions

Block PADP_L10 analyzes all events affecting a PA field device acyclically. It generates the slot-specific MODE (PA_MODE (Page 397)) and the value status for signal processing blocks. The permitted PA_MODE is defined for PA field devices.

If you are using modular PA field devices (DPV0), the next block (MOD_PAX0/MOD_PAL0) always reports the events in slot 0 of the PA field device in the DP/PA link. The affected MOD_PAX0/MOD_PAL0 block is enabled.

## How it works

Block PADP_L10 is enabled to run by the higher-level OB_DIAG1 block. The event to be evaluated is entered in the start information (CPU_DIAG) of OB_BEGIN. Block PADP_L10 checks the geographic address and the number of slots (SLOT_NO) of the PA field device to determine whether it is responsible for this event.

For a diagnostic event (OB 82, OB 55, OB 56, OB 57), SFB 54 is used to synchronously read the data from OB_BEGIN.

If diagnostic data could not be read synchronously from OB_BEGIN or when requested by OB_DIAG1 (EN_DIAG = TRUE), SFB 52 (RDREC) is used to read the current diagnostic data asynchronously.

Byte 9 of the additional alarm information contains the slot number of the field device that triggered the diagnostic interrupt. The corresponding slot is enabled.

The following diagnostic data is interpreted as a higher-level error in the block:

### Additional alarm information

| Byte no. | DPV1 name | Bit no. | Value | Info |
|---|---|---|---|---|
| Byte 1 to 6 | | | DDLM_SLAVE_DIAG | |
| Byte 7 | Header | Bit 7<br>Bit 6<br>Bit 5 to bit 0 | 0<br>0<br>8 or optional | fixed<br>fixed<br>Length of diagnostic data |
| Byte 8 | Status_Type | Bit 7<br>Bit 6 to bit 0 | 1<br>126 | Status<br>Highest vendor-specific status, will not be used in future |
| Byte 9 | Slot_number | | Slot number of the PB | The PB contains the diagnostic data. |
| Byte 10 | Specifier | Bit 7 to bit 2<br>Bit 1 to bit 0 | Reserved<br>1: status is displayed<br>2: status is not displayed | depends on the content of diagnostic data |
| Byte 11 to 14 | | | Diagnostics | |
| optional<br>Bytes 11 to 20 | | | | |

With a DPV0 PA field device, the diagnostic data is always assigned to slot 0.

It is also possible to generate slot-specific diagnostics for DPV1 PA field devices. These have not yet been defined. With a DPV1 field device, only the field-device slot that triggered the diagnostics is enabled. Evaluation of the coding in the context of higher-level errors in the slot-specific OMODE_xx (Page 396) outputs is based solely on the diagnostic information at slot 0.

## MODE setting for PA profiles (PA_MODE)

For more information, refer to the section: "MODE settings for PA devices (Page 397)".

The function writes MODE_xx to the low word of the OMODE_xx (Page 396) output parameter. This occurs only during startup or if you set ACC_MODE = TRUE. The current slot value status is written to the most significant byte. If the result is positive, the system sets OMODE_xx = 16#80xxxxxx. The following events lead to the value status "invalid value due to higher-level error" (OMODE_xx = 16#40xxxxxx):

- Rack failure (OB 86) (output parameter QRACKF = TRUE)
- Failure/return of a field device (OB 83)
- Slot-specific diagnostic interrupt (OB82)

Bytes 11 to 14 of the additional interrupt information are evaluated to form the slot-specific value status:

| Byte | Bit | Mnemonics | Description | Display class |
|---|---|---|---|---|
| 11 | 0 | DIA_HW_ELECTR | Electronic hardware failure | R |
| | 1 | DIA HW MECH | Mechanical hardware failure | R |
| | 2 | DIA_TEMP_MOTOR | Excess motor temperature | R |
| | 3 | DIA TEMP ELECTR | Excess temperature at electronic circuit | R |
| | 4 | DIA MEM CHKSUM | Memory error | R |
| | 5 | DIA_MEASUREMENT | Measurement failure | R |
| | 6 | DIA NOT INIT | Device not initialized<br>(no auto-calibration) | R |
| | 7 | DIA_INIT_ERR | Auto-calibration error | R |
| 12 | 0 | DIA ZERO ERR | Zero error (limit position) | R |
| | 1 | DIA_SUPPLY | No power supply (electr. pneum.) | R |
| | | | Invalid configuration | |
| | 2 | DIA CONV INVAL | Warm start executed | R |
| | 3 | DIA_WARMSTART | Complete restart executed | A |
| | 4 | DIA COLDSTART | Maintenance required | A |
| | 5 | DIA MAINTENANCE | Invalid identifier | R |
| | 6 | DIA_CHARACT | = 1, if the ID number of | R |
| | 7 | IDENT NUMBER Violation | the current cyclic data transfer<br>and the value of the IDENT NUMBER<br>parameter of the physical block are different | R |

| Byte | Bit | Mnemonics | Description | Display class |
|---|---|---|---|---|
| 13 | 0 | DIA_MAINTENANCE_ALARM | Device error | R |
| | 1 | DIA_MAINTENANCE_DEMANDED | Maintenance requested | R |
| | 2 | DIA_FUNCTION_CHECK | Device is in function test or in simulation or is under local operator control (maintenance) | R |
| | 3 | DIA_INV_PRO_COND | The process conditions do not permit valid values to be returned. (Set if quality "Uncertain, process related, no maintenance" or "Bad, process related, no maintenance") | R |
| | 4...7 | Reserved | Reserved for PNO, default 0 | |
| 14 | 0...4 | Reserved | Reserved for PNO use | |
| | 5 | PROFILE_SPECIFIC_EXTENSION _AVAILABLE | = 0: For devices with this profile | |
| | 6 | MANUFACTURER_SPECIFIC _EXTENSION_AVAILABLE | = 0: For devices with this profile | |
| | 7 | EXTENSION_AVAILABLE | = 0: No further diagnostic information available<br>= 1: Further diagnostic information available in DIAGNOSIS_EXTENSION | |

Display class R = incoming / outgoing events.

Display class A = incoming events that are reset by the field device after a few cycles. All events in byte 11 and the events of bit 0, 1 and 2 in byte 12 lead to the value status “Higher-level error” (OMODE_xx (Page 396) = 16#40xxxxxx).

Bytes 11, 12, 13, and 14 are entered in bytes 0 to 3 of the PA_DIAG parameter to generate messages and MS in the MOD_PAX0/MOD_PAL0 block.

## Special features of PA_AO and PA_DO field devices

With the field devices listed above, the PA profiles can be defined at two different slots. In this case, the driver generator will assign the mode code of the PA field device to the first slot and the mode code 16#8000 to the second at the corresponding MODE inputs of the block. The diagnostic information of the first and of the second slot are linked by a logical OR operation, which allows the generation of a uniform value status for the PA_x block.

## Redundancy

The higher-level block evaluates the redundancy of DP master systems operating in an H system.

## OMODE Structure

You will find additional information in the "OMODE (Page 396)" section.

## Addressing

You will find additional information in the "Addressing (Page 399)" section.

## Error handling

The plausibility of input parameters is not checked.

## Startup characteristics

A restart (OB100) is reported via the LSB in byte 2 of the OMODE_xx (Page 396) outputs.

## Time response

Not available

## Message response

Not available

## Operator control and monitoring

The block has no faceplate.

## 3.39.2 I/Os of PADP_L10

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O | Meaning | Type | De-fault |
|---|---|---|---|
| **DADDR** | Diagnostic address of the PA field device or DP/PA link | INT | 0 |
| EN_DIAG | 1 = Queued diagnostic event | BOOL | 0 |
| MODE_xx | Mode slot (xx = 16 - 31) | WORD | 0 |
| **PADP_ADR** | Address of the PA field device | BYTE | 0 |
| PROF_V30 | 1 = PA slave profile V3.0 | BOOL | 0 |
| **RACK_NO** | Address of PA field device or rack number | BYTE | 255 |
| **SLOT0_NO** | Slot number 0 of the field device in the DP/PA or Y link, or 0 if there is no link | BYTE | 0 |
| **SLOTS_NO** | Number of slots of the field device | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O | Meaning | Type | De-fault |
|---|---|---|---|
| **DINFO** | Diagnostic status of the PA field device | STRUCT | |
| EN_M_xx | Enable slot (xx = 0 - 31) | BOOL | 0 |
| OMODE_xx | Mode slot (xx = 16 - 31) | DWORD | 0 |
| **PA_DIAG** | PA field device diagnostic information | DWORD | 0 |
| QERR | 1 = program error | BOOL | 0 |
| **QMODF** | 1 = field device error/fault | BOOL | 0 |
| **QPERAF** | 1 = I/O access error | BOOL | 0 |
| **QRACKF** | 1 = (Link) rack/station error | BOOL | 0 |

## In-out parameters

| I/O | Meaning | Type | De-fault |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | CPU diagnostics | STRUCT | |
| CPU_OB_5X | OB_5x startup information | STRUCT | |
| RAC_DIAG | PA field device or link diagnostics | STRUCT | |

# 3.40 PDM_MS: Monitoring of the maintenance status

## 3.40.1 Description of PDM_MS

### Object name (type + number)

FB 81

- I/Os of PDM_MS (Page 308)

### How it works

The block reports the Maintenance State supplied by PDM via the Maintenance Station.

### Additional information

You will find more detailed information in the section Message texts and associated values of PDM_MS (Page 309).

### 3.40.2 I/Os of PDM_MS

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: General Information About Block Description (Page 9).

#### Input parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number (x = 1, 2) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| RUNUPCYC | Number of replacement run cycles | INT | 3 |
| CYCL_UPD | 1 = Cyclic update active | BOOL | 1 |

#### Output parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| MSGSTATx | Message error information (x = 1, 2) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |

#### Additional information

Additional information is available in the section:

Message texts and associated values of PDM_MS (Page 309)

### 3.40.3 Message texts and associated values of PDM_MS

#### Messaging

The statuses are generated with ALARM_8P for messages requiring acknowledgment, and with NOTIFY_8P for those not requiring acknowledgment. The message function can be disabled by setting EN_MSG = 0. In this case MS = 8 is set.

#### Assignment of message text and message class

The process control messages of ALARM_8P with ED_ID1 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | Bad, maintenance alarm | S |
| | 2 | | No message |
| | 3 | Uncertain, maintenance request | F |
| | 4 | Good, maintenance required | M |
| | 5 | Bad, device out of service | S |
| | 6 | | No message |
| | 7 | | No message |
| | 8 | | No message |

The process control messages of ALARM_8P with ED-ID2 are assigned as follows:

| Message block NOTIFY_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID2 | 1 | bad, passivated | SA |
| | 2 | | No message |
| | 3 | Bad, local operation/functional check | SA |
| | 4 | uncertain, simulation | SA |
| | 5 | Configuration change | SA |
| | 6 | | No message |
| | 7 | | No message |
| | 8 | | No message |

#### Additional information

Additional information is available in the section: Message Classes (Page 401)

# 3.41 PO_UPDAT: Output Process Image

## 3.41.1 PO_UPDAT: Output Process Image

### Object name (type + number)

FC 279

### Area of application

The PO_UPDAT block safeguards the output module functions "Hold last value" and "Apply substitute value" when a CPU is restarted (OB 100).

### Run Sequence

With the "Generate module drivers" CFC function, the PO_UPDAT block is automatically installed in OB 100 at the end.

### Description of Functions

On a CPU restart (OB 100), the CH_DO and CH_AO blocks write the start values to the process image. The PO_UPDAT block sends all process images (partitions) to the modules at the end of OB 100 in order for these values to be active immediately when the CPU goes into RUN. Output PO_MAP indicates the process image partitions which have been updated or are used in the system (BIT0: Process image 0, BIT15: Process image partition 15).

# 3.42 PS: Power supply monitoring

## 3.42.1 Description of PS

### Object name (type + number)

FB 89

- PS block I/Os (Page 314)

### Area of application

The PS block monitors the status of a rack power supply, and reports the associated error events.

### Calling OBs

The PS block must be installed in the run sequence of the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 81 | Power supply error |
| OB 83 | Remove/insert interrupt |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The block is installed in the run sequence downstream of the RACK block.
- The SLOT_NO input (slot number of the power supply) is configured.
- The EN input is interconnected with the output of an AND block.
  Its inputs are interconnected with the output EN_SUBx of the OB_BEGIN block, the output EN_Rxxx of the SUBNET block and the output EN_Mxx of the RACK block.
- The OUT structures CPU_DIAG of the OB_BEGIN block and RAC_DIAG of the RACK block are interconnected with the IN_OUT structures of the same name of the PS block.

## Function and method of operation

The PS block reports events of the power supply error OB 81 and OB 83 relating to the power supply module. The module is installed to supply power to the central rack, and to each expansion rack.

---

Note

Note the following:

- If a battery fails, the battery must always be replaced with the power supply turned on. Then press the "FMR" button. In all other situations, the block does not reset a reported error.
- For redundant power supply modules in a rack with a standard CPU, a corresponding message is sent for both power supply modules in the event of a battery error or power supply error. You can tell which module is affected by the "BATTF" LED that lights up.

---

## Redundancy

In a redundant system, the block is also installed extra for the power supply of the redundant rack.

## Error handling

The error handling of the block is limited to the evaluation of the error information of ALARM_8P.
Refer to the section "Error information of output parameter MSG_STAT (Page 398)" for additional information on error handling.

## Startup characteristics

The PS block initializes the messages of the ALARM_8P.

## Overload behavior

Not available

## Time response

You will find additional information in the "Message response" section.

## Message response

After OB 81 or OB 83 is called, the block analyzes the status of the power supply of the rack assigned to it. It generates the "Backup battery failure", "Backup voltage failure" and "24 V supply failure" or "Module removed" or "Wrong or faulty module" messages with ALARM_8P. The message function can be disabled by setting EN_MSG = FALSE.

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of PS (Page 315)

Maintenance status of MS (Page 403)

### 3.42.2 I/Os of PS

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| SLOT_NO | Slot number of the power supply | BYTE | 0 |

#### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |

#### In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **CPU_DIAG** | CPU diagnostics (system structure) | STRUCT | |
| **RAC_DIAG** | System structure | STRUCT | |

#### Additional information

For additional information, refer to the following sections:

Message texts and associated values of PS (Page 315)

Maintenance status of MS (Page 403)

### 3.42.3 Message texts and associated values of PS

#### Assignment of message text and message class

| Message No. | Default message text | Message class |
|---|---|---|
| 1 | @1%d@/ @2%d@/ @3%d@: Backup battery failure | M |
| 2 | @1%d@/ @2%d@/ @3%d@: Backup voltage failure | M |
| 3 | @1%d@/ @2%d@/ @3%d@: 24 V power supply failure | M |
| 4 | Module @1%d@/@2%d@/@3%d@: Removed | S |
| 5 | Module @1%d@/@2%d@/@3%d@: wrong or faulty | S |
| 6 | | No message |
| 7 | | No message |
| 8 | | No message |

#### Assignment of Associated Values

| Associated value | Block parameters |
|---|---|
| 1 | DP master system ID of the rack of the power supply (RAC_DIAG.SUBN_ID) |
| 2 | Rack number of the power supply (RAC_DIAG.RACK_NO) |
| 3 | Slot number of the power supply (SLOT_NO) |

#### See also

Message Classes (Page 401)

# 3.43 RACK: Rack monitoring

## 3.43.1 Description of RACK

### Object name (type + number)

FB 107

- RACK block I/Os (Page 320)

### Area of application

The RACK block monitors the status of a rack, and reports the associated error events.

### Calling OBs

The block is installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 70 | I/O redundancy error |
| OB 72 | CPU redundancy error |
| OB 81 | Power supply error |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The RACK block is installed in the run sequence downstream of the SUBNET block.
- The RACK_NO, DADDR, SUBN1_ID, SUBN2_ID, and SUBN_TYP inputs are configured.
- The EN input is interconnected with the output of an AND block.
  whose inputs are interconnected to the EN_SUBx output of the OB BEGIN block, and to the EN_Rxxx output of the SUBNET block.
- The CPU_DIAG OUT structure of the OB_BEGIN block and SUB_DIAG of the SUBNET block are interconnected with the IN_OUT structures of the same name of the RACK block.

## Function and method of operation

The RACK block generates a process control error message for the OS in the event of redundancy losses and rack/station failures. It also indicates internal errors of the rack/station (SUBN1ERR, SUBN2ERR), and of the preferred channel (SUBN1ACT, SUBN2ACT) if there are active DP slaves at its outputs. The output structure RAC_DIAG contains the geographic address of the rack, and the group error information RACK_ERR. If RACK_ERR = 1, the corresponding rack is not available.

The block is installed in the OBs listed above once at each station or local I/O device. The SUBNET block enables the runtime group that contains the RACK block. Start and diagnostic information is read from the CPU_DIAG IN_OUT structure, which is interconnected with the CPU_DIAG structure of the OB_BEGIN block. The RACK block has one enable output for each rack (station) slot.

The RACK block generates the number of a corresponding message (see "Message Response") on the basis of the start information of the calling OBs if the current block is affected.

The block evaluates error events, and uses the diagnostic address DADDR of the DP slave to determine the currently active preferred channel (SUBN1ACT, SUBN2ACT) of redundant PROFIBUS DP interface circuits.

Note: If you want to change the SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs online, you must set input ACC_ID = TRUE. to update the output values.

## Redundancy

In H systems with distributed I/Os, the RACK block supports redundancy of the DP Master systems. If you want to use this function, you must configure the SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs of the RACK block with the numbers of the redundant DP master systems. If there is no redundancy, the remaining input must be set to the (default) value 16#FF.

---

**Note**

With redundant CPU racks, the two RACK blocks inserted in the system chart are only responsible for enabling lower-level block chains. Their maintenance status MS is therefore irrelevant. The "Good" and "Not redundant" state is always shown in the associated faceplate and block icon because the bits 0 to 16 of the MS are always "0" in this case.

---

## Error handling

Error handling of the block is limited to evaluation of the error information of ALARM_8P.

You will find more information about this in "Error information of output parameter MSG_STAT (Page 398)".

## Startup characteristics

The RACK block initializes ALARM_8P messages. It checks availability of the station and, in H systems, determines the preferred channel of the station.

The SUB_DIAG.V1_MODE structure (0 = compatibility mode, 1 = DPV1 mode) is transferred to the RAC_DIAG.V1_MODE structure.

## Overload behavior

The RACK block counts the OB 86 calls (except in the case of a DP master system failure; see SUBNET block). The counter is reset in OB 1. If more than two OB 86 events occur in succession before the cycle control point (OB 1) is reached, these will be rejected and the message "Station...: Multiple failure" is output. When an OB 86 call is rejected, the rack (station) is registered as having failed.

## Time response

See "Message Response"

## Message response

After it is called by OB 70, OB 72, OB 85 or OB 86, the block analyzes the status of its assigned CPU, DP master and DP slave. If the rack (station) loses redundancy or fails, the block outputs the corresponding messages by broadcasting an ALARM_8P. The message function can be disabled by setting EN_MSG = FALSE.

The block generally reports only the events that were originally generated in the rack that it monitors. Redundancy loss and station failures which are caused by the failure of a DP master or CPU are initially neither signaled nor indicated at the SUBN1ERR and SUBN2ERR outputs.

The DELAY input is used to delay the outputting of error messages for an outgoing, higher-priority error. For example, if the RACK block recognizes an outgoing error at an interconnected DP master, it initially assumes that there is a faulty assigned DP slave in the rack it monitors, and sets the corresponding output SUBNxERR. The error status is not reset until the DP slave returns (in this case: OB 86, OB 70). The RACK blocks suppress the potential slave failure states for DELAY seconds so as not to trigger a surge of messages from DP slaves which are not yet synchronized when the master returns. An error message is only output to the OS when the DP slave has not reported its return before this delay time has expired.

Note: Do not set the value of DELAY too high, otherwise DP slaves that were removed during the master failure or are defective will be signaled to the OS too late after the DP master returns.

The RACK block generates the following messages in the OBs listed below:

| OB | Start event | Message |
|---|---|---|
| OB 1 | Cyclic processing | Repeat the update of ALARM_8P outputs/messages, if necessary |
| OB 70 | Redundancy loss | Station redundancy loss/return |
| OB 81 | Power supply error | |
| OB 85 | Program execution error | Station failure, incoming/outgoing |
| OB 86 | Rack failure | Station failure, incoming/outgoing |
| OB 100 | Restart | Initialization of ALARM_8P |

## Operator control and monitoring

**Note:** If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of RACK (Page 322)

Maintenance status of MS (Page 403)

## 3.43.2 I/Os of RACK

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **DADDR** | Diagnostic address of the DP slave | INT | 0 |
| DELAY | Interrupt delay (s) | INT | 15 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_Mxx | 1 = Enable module xx (xx = 0 - 63) | BOOL | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| **RAC_DIAG** | System structure | STRUCT | |
| **SUBN1ACT** | 1 = Slave 1 is active | BOOL | 0 |
| **SUBN1ERR** | 1 = Error in slave 1 | BOOL | 0 |
| **SUBN2ACT** | 1 = Slave 2 is active | BOOL | 0 |
| **SUBN2ERR** | 1 = Error in slave 2 | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | CPU diagnostics (system structure) | STRUCT | |
| SUB_DIAG | OB_Start information | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of RACK (Page 322)

Maintenance status of MS (Page 403)

### 3.43.3 Message texts and associated values of RACK

#### Assignment of message text and message class

| Message no. | Default message text | Message class |
|---|---|---|
| 1 | Station @1%d@/ @3%d@: Redundancy loss | F |
| 2 | Station @2%d@/ @3%d@: Redundancy loss | F |
| 3 | Station @1%d@/ @3%d@: Failure | S |
| 4 | Station @2%d@/ @3%d@: Failure | S |
| 5 | | No message |
| 6 | | No message |
| 7 | | No message |
| 8 | Station @1%d@/ @3%d@: Multiple failure | S |

#### Assignment of associated values

| Associated value | Block parameters |
|---|---|
| 1 | ID of the primary DP master system (SUBN1_ID) |
| 2 | ID of the redundant DP master system (SUBN2_ID) |
| 3 | Rack/station number (RACK_NO) |

#### See also

Message Classes (Page 401)

# 3.44 RED_F: Status processing of redundant F modules

## 3.44.1 Description of RED_F

### Object name (type + number)

FC 289

- RED_F block I/Os (Page 325)

### Area of application

The RED_F block is used to set up redundant F modules in safety mode.

### Calling OBs

The block must be installed in the same OB before the OR block. It is also installed in OB 100.

### Use in CFC

The following actions are executed automatically with the "Generate module drivers" CFC function:

- The RED_F block is installed before the OR block in its OB.
- MODE1_xx inputs are interconnected with the OMODE_xx outputs of the MOD_x block in the primary module.
- MODE2_xx inputs are interconnected with the OMODE_xx outputs of the MOD_x block in the redundant module.
- The RACKF1 input is interconnected with the QRACKF output of the MOD_x block of the primary module.
- The RACKF2 input is interconnected with the QRACKF output of the MOD_x block of the redundant module.
- The MS1 input is interconnected with the O_MS output of the MOD_x block of the primary module.
- The MS2 input is interconnected with the O_MS output of the MOD_x block of the redundant module.
- The ACTIV_H and ACTIV_L outputs are interconnected with the inputs with the same name in the OR block.
- The CH_INF_H and CH_INF_L outputs are interconnected with the inputs with the same name in the OR block.

- The RETURN_VAL output is interconnected with the RED_STAT input of the OR block.
- The MODUL_STATUS_WORD output is interconnected with the MOD_STAT input of the OR block.

## Function and method of operation

The RED_F block processes the status of all channels cyclically based on the outputs OMODE_xx or the MOD_x blocks and then forms the information on redundancy of the OR blocks.

## Addressing

Not available

## Error handling

Not available

## Startup characteristics

Not available

## Time response

Not available

## Message response

Not available

## Operator control and monitoring

The block has no faceplate.

### 3.44.2 I/Os of RED_F

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name in **bold** characters = I/O is visible; I/O name in standard characters = I/O is hidden.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| MODE1_xx | Channel mode (xx = 00 – 31) on the primary module | DWORD | 0 |
| MODE2_xx | Channel mode (xx = 00 – 31) on the redundant module | DWORD | 0 |
| MS1 | Maintenance status (MS) 1 | BOOL | 0 |
| MS2 | Maintenance status (MS) 2 | BOOL | 0 |
| **RACKF1** | 1 = error rack 1 | BOOL | 0 |
| **RACKF2** | 1 = error rack 2 | BOOL | 0 |

#### Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **ACTIV_H** | 1 = module with more significant address is active | BOOL | 0 |
| **ACTIV_L** | 1 = module with less significant address is active | BOOL | 0 |
| CH_INF_H | 1 = channel with more significant address is active | DWORD | 0 |
| CH_INF_L | 1 = channel with less significant address is active | DWORD | 0 |
| **MODUL_STATUS_WORD** | Status information | WORD | 0 |
| **RETURN_ VAL** | Error information | INT | 0 |

#### Additional information

You will find more information in:

Maintenance status of MS (Page 403)

# 3.45 PROFINET blocks

## 3.45.1 OB_BEGIN_PN: CPU diagnostics

### 3.45.1.1 Description of OB_BEGIN_PN

#### Object name (type + number)

FB 130

- I/Os of OB_BEGIN_PN (Page 326)

#### Area of application

Block OB_BEGIN_PN is used for CPU diagnostics of the automation system (AS). By installing the block in CFC, the system creates all acyclic run sequences (OBs) in which the driver blocks of PCS 7 Advanced Process Library are executed.

### 3.45.1.2 I/Os of OB_BEGIN_PN

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number ALARM_8P_x (x = 1 - 4, assigned by the ES) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| SUB0IDxx | DP master system 1 IDxx (xx = 00 - 14)/PN IO system 1 (100-115) | BYTE | 255 |
| SUB1IDxx | DP master system 2 IDxx (xx = 00 - 14)/PN IO system 2 (100-115) | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CPU_DIAG | System structure: CPU diagnostics | STRUCT | |
| CPU_DIAG_PN | System structure: CPU diagnostics (PN) | | |
| CPU_OB_4X | Start information OB 40 - OB 47 | STRUCT | |
| CPU_OB_5X | Start information OB 55, OB 56, OB 57 | STRUCT | |
| CPU_OB_5X_PN | Start information OB 55, OB 56, OB 57 | STRUCT | |
| CPU_OB_6X | Start information OB 60 - OB 64 | STRUCT | |
| CPUERR_0 | 1 = CPU error in rack 0 *) | BOOL | 0 |
| CPUERR_1 | 1 = CPU error in rack 1 *) | BOOL | 0 |
| EN_SUBx | Enable SUBNET x (DP: x = 0 - 14/PN: x = 100-115)) | BOOL | 0 |
| MASTER_0 | 1 = Master CPU in rack 0 | BOOL | 0 |
| MASTER_1 | 1 = Master CPU in rack 1 | BOOL | 0 |
| MSGSTATx | STATUS output of ALARM_8P_x (x = 1 - 5) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QERR | 1 = processing error | BOOL | 1 |
| QMSGERx | Error output of ALARM_8P_x (x = 1 - 5) | BOOL | 0 |
| SZL_71 | System structure: SZL71 | STRUCT | |

The structure of the CPU_DIAG is integrated as OUT in the OB_BEGIN block, and as IN_OUT in all other blocks with this I/O.

*) You will find additional information about CPU errors in the CPU Manual.

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of OB_BEGIN_PN (Page 328)

Maintenance Status of MS (Page 403)

### 3.45.1.3 Message texts and associated values of OB_BEGIN_PN

#### Assignment of message text and message class

Process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message block ALARM_8P | Message number | OB no. | Default message text | Message class |
|---|---|---|---|---|
| EV_ID1 | 1 | OB 85 | OB @7%d@ not loaded | S |
| | 2 | | | No message |
| | 3 | OB 84 | Interface error | S |
| | 4 | - | Error installing OB_BEGIN_PN/OB_END: No OB@2%d@ processing of stack @1%d@ | S |
| | 5 | OB 85 | Program execution error: @7%d@: @10%2s@@8%d@/@9%d@ | S |
| | 6 | OB 122 | I/O read access error: @4%2s@@5%d@ Address: @6%d@ | S |
| | 7 | OB 122 | I/O write access error: @4%2s@@5%d@ Address: @6%d@ | S |
| | 8 | OB 84 | Performance of an H-Sync coupling impaired | S |

Messages 1, 4, 5, 6, 7 and 8 are only incoming events. They are reset to "outgoing" status during the normal run sequence (OB 1) of the block.

#### Associated values of ALARM_8P with EV_ID1

Process control messages are generated with six associated values at EV_ID1 via ALARM_8P. The table below shows how the associated values are assigned to the block parameters.

| Message block ALARM_8P | Associated value | Block parameter | Data type |
|---|---|---|---|
| EV_ID1 | 1 | CPU_DIAG_PN.OB_S_NUM_CNT | BYTE |
| | 2 | TINFO_TOP_SI_NUM | BYTE |
| | 3 | OB 72_supp_info 1 | WORD |
| | 4 | OB 122_BLK_TYP | WORD |
| | 5 | OB 122_BLK_NUM | WORD |
| | 6 | OB 122_MEM_ADDR | WORD |
| | 7 | OB 85_supp_info 1 | WORD |
| | 8 | OB 85_HW_supp_info 2_3 | WORD |
| | 9 | OB 85_LW_supp_info 2_3 | WORD |
| | 10 | OB 85_DKZ2_3 | WORD |

## The process control messages of ALARM_8P with EV_ID2 are assigned as follows:

| Message block ALARM_8P | Message number | OB no. | Default message text | Message class |
|---|---|---|---|---|
| EV_ID2 | 1 | OB 80 | Net consumption of all OBs exceeds max. limit | M |
| | 2 | OB 80 | Emergency operation, cyclic OBs are used | S |
| | 3 | OB 80 | Priorities of the cyclic OBs not PCS 7 conform | M |
| | 4 | OB 84 | Memory error detected and corrected by operating system. | S |
| | 5 | OB 84 | Accumulation of detected and corrected memory errors | S |
| | 6 | OB 84 | Error in PC operating system | S |
| | 7 | OB 121 | Programming error @1%d@: @2%2s@@5%d@ /@6%d@/@4%d@/@3%d@ | S |
| | 8 | OB 84 | Multiple-bit memory error detected and corrected | S |

Messages 1 to 3 are generated in CPU_RT and forwarded to OB_BEGIN_PN.

Messages 4, 5, 7 and 8 are only incoming events. They are reset to "outgoing" status during the normal run sequence (OB 1) of the block.

Message 7 is to be interpreted as follows, in accordance with the error code number before the colon:

OB 121_BLK_TYP/OB 121_BLK_NUM/OB 121_PRG_ADDR/OB 121_FLT_REG/OB 121_RESERVED_1.

## Associated values of ALARM_8P with EV_ID2

Process control messages are generated with six associated values at EV_ID2 via ALARM_8P. The table below shows how the associated values are assigned to the block parameters.

| Message block ALARM_8P | Associated value | Block parameter | Data type |
|---|---|---|---|
| EV_ID2 | 1 | OB 121_SW_FLT | BYTE |
| | 2 | OB 121_BLK_TYP | WORD |
| | 3 | OB 121_RESERVED_1 | BYTE |
| | 4 | OB 121_FLT_REG | WORD |
| | 5 | OB 121_BLK_NUM | WORD |
| | 6 | OB 121_PRG_ADDR | WORD |

### Process control messages of ALARM_8P with EV_ID3 are assigned as follows:

| Message block ALARM_8P | Message number | OB no. | Default message text | Message class |
|---|---|---|---|---|
| EV_ID3 | 1 | OB 80 | Cycle time exceeded: @1%d@ms OB@2%d@ | S |
| | 2 | OB 80 | OB request: OB3x still being processed | F |
| | 3 | OB 80 | TOD interrupt OB @3%d@ expired (TOD jump) | S |
| | 4 | OB 80 | TOD interrupt OB @4%d@ expired (Stop/Run) | S |
| | 5 | OB 80 | OB request: Overflow PRIO @5%d | S |
| | 6 | OB 80 | Clocked interrupt timeout: OB@6%d@ PRIO @7%d@ | S |
| | 7 | OB 80 | Interrupt lost: OB@8%d@ PRIO @9%d@ | S |
| | 8 | OB 80 | CiR synchronization time: @10%d@ ms | S |

Message 2 is generated in CPU_RT (Page 23) and forwarded to OB_BEGIN.

Messages 1 to 8 are only incoming events. They are reset to "outgoing" status during the normal run sequence (OB 1) of the block.

### Associated Values of ALARM_8P with EV_ID3

Process control messages are generated with seven associated values at EV_ID3 via ALARM_8P. The table below shows how the associated values are assigned to the block parameters.

| Message block ALARM_8P | Associated value | Block parameter | Data type |
|---|---|---|---|
| EV_ID3 | 1 | Cycle time (OB 80_supp_info 1) | WORD |
| | 2 | Cause OB (OB 80_1st byte supp_info 2_3) | BYTE |
| | 3 | Cycle time (OB 80_supp_info 1) | WORD |
| | 4 | Cycle time (OB 80_supp_info 1) | WORD |
| | 5 | Priority class (OB 80_2nd byte supp_info 2_3) | BYTE |
| | 6 | Cause OB (OB 80_1st byte supp_info 2_3) | BYTE |
| | 7 | Priority class (OB 80_2nd byte supp_info 2_3) | BYTE |
| | 8 | Cause OB (OB 80_1st byte supp_info 2_3) | BYTE |
| | 9 | Priority class (OB 80_2nd byte supp_info 2_3) | BYTE |
| | 10 | Cycle time (OB 80_supp_info 1) | WORD |

## Process control messages for ALARM_8P with EV_ID4 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID_4 | 1 | OB 88(@6W%t#OB_BEGIN_PN_TXT@): OB@1%d@ PRIO@2%d@ @3%2s@@4%d@ /@5%d@ | S |
| | 2 | OB_BEGIN_PN: Diagnostics error RALRM STATUS = @7%8X@ | S |
| | 3 | | No message |
| | 4 | | No message |
| | 5 | | No message |
| | 6 | | No message |
| | 7 | | No message |
| | 8 | | No message |

## Associated values of ALARM_8P with EV_ID4

| Message block ALARM_8P | Associated value | Meaning |
|---|---|---|
| EV_ID4 | 1 | Cause OB (M_OB 88.FLT_OB) |
| | 2 | Priority class (M_OB 88.FLT_OB_PRIO) |
| | 3 | Block type (M_OB 88.BLK_TYP) |
| | 4 | Block number (M_OB 88.FLT_NUM) |
| | 5 | MC7 command causing error<br>Relative address (M_OB 88.FLT_ADDR) |
| | 6 | Error number in OB_BEGIN_TXT (M_OB 88.T_OB 88) |
| | 7 | Status RALRM |

For additional information, see chapter: Message Classes (Page 401).

## 3.45.2 SUBNET_PN: Reduction of acyclic OB processing times

### 3.45.2.1 Description of SUBNET_PN

#### Object name (type + number)

FB 82

- I/Os of SUBNET_PN (Page 332)

#### Area of application

The SUBNET_PN block is used to reduce acyclic OB processing times. Only the blocks that are actually affected can be called in the case of an acyclic event.

### 3.45.2.2 I/Os of SUBNET_PN

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| **DADDR** | Diagnostic address of the PN IO device - primary subnet | INT | 0 |
| **DADDR_1** | Diagnostic address of the PN IO device - - redundant subnet | INT | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number ALARM_8P (assigned by the ES) | DWORD | 0 |
| EV_ID1 | Message number ALARM_8P (assigned by the ES) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| PNIO_ADR | Diagnostic address of PN IO interface - primary subnet | INT | 0 |
| PORT1_ADR | Diagnostic address of PORT1 - primary subnet | INT | 0 |
| PORT2_ADR | Diagnostic address of PORT2 - primary subnet | INT | 0 |
| PORT2_CONNECT | Port2 connection status - primary subnet | BYTE | 0 |
| PNIO_ADR_1 | Diagnostic address of PN IO interface - redundant subnet | INT | 0 |
| PORT1_ADR_1 | Diagnostic address of PORT1 – redundant subnet | INT | 0 |
| PORT1_CONNECT_1 | Port1 connection status – redundant subnet | BYTE | 0 |
| PORT2_ADR_1 | Diagnostic address of PORT2 – redundant subnet | INT | 0 |
| PORT2_CONNECT_1 | Port2 connection status – redundant subnet | BYTE | 0 |

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| SUBN_TYP | 1 = external DP interface | BOOL | 0 |
| SUBN1_ID | ID of the primary DP master system | BYTE | 255 |
| SUBN2_ID | ID of the redundant DP master system | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| EN_Rxxx | 1 = Enable rack (xxx = 0 - 255) | BOOL | 0 |
| MASTER_0 | 1 = Master CPU in rack 0 | BOOL | 0 |
| MASTER_1 | 1 = Master CPU in rack 1 | BOOL | 0 |
| MSGSTATx | STATUS output of ALARM_8P_x (x = 1 - 2) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QMSGERx | ERROR message of ALARM_8P_x (x = 1 - 2) | BOOL | 0 |
| SUB_DIAG | System structure: CPU diagnostics | STRUCT | |
| SUBN1ERR | 1 = Error in DP master system 1 | BOOL | 0 |
| SUBN2ERR | 1 = Error in DP master system 2 | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | CPU diagnostics system structure | STRUCT | |
| CPU_OB_5X_PN | Start information OB 55, OB 56, OB 57 | STRUCT | |
| SZL_71 | System structure SZL71 | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of SUBNET_PN (Page 334)

Maintenance Status of MS (Page 403)

### 3.45.2.3 Message texts and associated values of SUBNET_PN

#### Assignment of message text and message class

The process control messages of ALARM_8P with EV_ID are assigned as follows:

| Message no. | Default message text | Message class |
|---|---|---|
| 1 | PN IO system @1%d@: Redundancy loss | F |
| 2 | PN IO system @2%d@: Redundancy loss | F |
| 3 | PN IO system @1%d@: Failure | S |
| 4 | PN IO system @2%d@: Failure | S |
| 5 | PN IO system @2%d@: Multiple failure | S |
| 6 | CPU loss of redundancy in rack @4%d@ | F |
| 7 | | No message |
| 8 | | No message |

Associated values of ALARM_8P with EV_ID

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the primary PN IO system (SUBN1_ID) | BYTE |
| 2 | ID of the redundant PN IO system (SUBN2_ID) | BYTE |
| 3 | ID of the multiple failure PN IO system | BYTE |
| 4 | CPU rack number | BYTE |

The process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message no. | Default message text | Message class |
|---|---|---|
| 1 | PN IO system @1%d@: Port1 error | F |
| 2 | PN IO system @1%d@: Port2 error | F |
| 3 | PN IO system @1%d@: Uncertain, maintenance demanded | F |
| 4 | PN IO system @1%d@: Good, maintenance required | M |
| 5 | PN IO system @2%d@: Port1 error | F |
| 6 | PN IO system @2%d@: Port2 error | F |
| 7 | PN IO system @2%d@: Uncertain, maintenance demanded | F |
| 8 | PN IO system @2%d@: Good, maintenance required | M |

Associated values of ALARM_8P with EV_ID1

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the primary PN IO system (SUBN1_ID) | BYTE |
| 2 | ID of the redundant PN IO system (SUBN2_ID) | BYTE |
| 3 | ID of the PN IO system | WORD |

For additional information, see chapter: Message Classes (Page 401).

## 3.45.3 RACK_PN: Rack monitoring

### 3.45.3.1 Description of RACK_PN

#### Object name (type + number)

FB 90

- I/Os of RACK_PN (Page 336)

#### Area of application

The RACK_PN block monitors the status of a rack, and reports the associated error events.

#### Additional information

For additional information, refer to the following sections:

Message texts and associated values of RACK_PN (Page 338)

Maintenance Status of MS (Page 403)

### 3.45.3.2 I/Os of RACK_PN

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **DADDR** | Diagnostics address of the PN IO device | INT | 0 |
| DELAY | Interrupt delay (s) | INT | 15 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number ALARM_8P (assigned by the ES) | DWORD | 0 |
| EV_ID1 | Message number ALARM_8P (assigned by the ES) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| PNIO_ADR | Diagnostics address of the PN-IO interface | INT | 0 |
| PORT1_ADR | Diagnostics address of PORT1 | INT | 0 |
| PORT1_CONNECT | Port1 connection status | BYTE | 0 |
| PORT2_ADR | Diagnostics address of PORT2 | INT | 0 |
| PORT2_CONNECT | Port2 connection status | BYTE | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| SUBN_TYP | 1 = external DP interface | BOOL | 0 |
| SUBN1_ID | ID of the primary DP master system | BYTE | 255 |
| SUBN2_ID | ID of the redundant DP master system | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_Mxx | 1 = Enable rack xx (xx = 0 - 63) | BOOL | 0 |
| MSGSTATx | Status output of ALARM_8P_x (x = 1-2) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QMSGERx | ERROR message of ALARM_8P_x (x = 1 - 2) | BOOL | 0 |
| RAC_DIAG | System structure | STRUCT | |
| SUBN1ACT | 1 = Slave 1 is active | BOOL | 0 |
| SUBN1ERR | 1 = Error in slave 1 | BOOL | 0 |
| SUBN2ACT | 1 = Slave 2 is active | BOOL | 0 |
| SUBN2ERR | 1 = Error in slave 2 | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | CPU diagnostics (system structure) | STRUCT | |
| SUB_DIAG | OB_Start information | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of RACK_PN (Page 338)

Maintenance Status of MS (Page 403)

### 3.45.3.3 Message texts and associated values of RACK_PN

#### Assignment of message text and message class

The process control messages of ALARM_8P with EV_ID are assigned as follows:

| Message no. | Default message text | Message class |
|---|---|---|
| 1 | Station @1%d@/ @3%d@: Redundancy loss | F |
| 2 | Station @2%d@/ @3%d@: Redundancy loss | F |
| 3 | Station @1%d@/ @3%d@: Failure | S |
| 4 | Station @2%d@/ @3%d@: Failure | S |
| 5 | | No message |
| 6 | | No message |
| 7 | | No message |
| 8 | Station @1%d@/ @3%d@: Multiple failure | S |

Associated values of ALARM_8P with EV_ID

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the primary PN IO system (SUBN1_ID) | BYTE |
| 2 | ID of the redundant PN IO system (SUBN2_ID) | BYTE |
| 3 | Rack/station number (RACK_NO) | BYTE |

The process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message no. | Default message text | Message class |
|---|---|---|
| 1 | Station @4%d@/ @3%d@: Port1 error | F |
| 2 | Station @4%d@/ @3%d@: Port2 error | F |
| 3 | Station @4%d@ @3%d@: Uncertain, maintenance demanded | F |
| 4 | Station @4%d@/ @3%d@: Good, maintenance required | M |
| 5 | | No message |
| 6 | | No message |
| 7 | | No message |
| 8 | | No message |

Associated values of ALARM_8P with EV_ID1

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the primary PN IO system (SUBN1_ID) | BYTE |
| 2 | ID of the redundant PN IO system (SUBN2_ID) | BYTE |
| 3 | Rack/station number (RACK_NO) | BYTE |
| 4 | ID of the PN IO system | WORD |

For additional information, see chapter: Message Classes (Page 401).

## 3.45.4 MOD_D8_PN: Monitoring S7-300 SM modules with up to 8 channels and with diagnostics functions

### 3.45.4.1 Description of MOD_D8_PN

#### Object name (type + number)

FB 197

- I/Os of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN (Page 340)

#### Area of application

Block MOD_D8_PN monitors S7-300 SM modules with a maximum of 8 channels and with diagnostics functions.

### 3.45.4.2 I/Os of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CHAN_NUM | Number of channels | INT | 0 |
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number ALARM_8P_x<br>(x = 1 - 2/1 - 3/1 - 4, assigned by the ES) | DWORD | 0 |
| FEATURE_xx | Feature parameter (xx = 01 - 04) | WORD | 0 |
| FEATURE_yy | Feature parameter (yy = 05 - 10) | DWORD | 0 |
| **LADDR** | Logic input address of the module | INT | 0 |
| **LADDR1** | Logical output address of the module (if output address is not the same as input address). | INT | 0 |
| MODE_xx | Operating mode channel xx (xx = 00 - 07/00 - 15/00 - 23) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| SUBSL_NO | Subslot number | BYTE | 0 |
| SUBN_TYP | 1 = external PN interface | BOOL | 0 |
| SUBN1_ID | PN IO system 1 ID (100 - 115) | BYTE | 255 |
| SUBN2_ID | PN IO system 2 ID (100 - 115) | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DIAG_INFx | System structure: Diagnostic information | STRUCT | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | 0 |
| EXT_STATx | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression active | DWORD | 16#00000000 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACKx | Message acknowledgment (x = 1 - 2/1 - 3/1 - 4) | WORD | 0 |
| MSGSTATx | Message error information (x = 1 - 2/1 - 3/1 - 4) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Operating mode channel xx (xx = 00 - 07/00 - 15/00 - 23) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack error | BOOL | 0 |

### In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

### Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN (Page 342)

Maintenance Status of MS (Page 403)

## 3.45.4.3 Message texts and associated values of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN

### Assignment of message text and message class

Process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | Module @1%d@/@2%d@/@3%d@/@6%d@: Removed | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@6%d@: Access error | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@6%d@: @5W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT@ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@6%d@: Multiple diagnostics interrupt | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@6%d@: @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@6%d@: @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@6%d@: @5W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT@ | F |
| | 8 | | No message |

Associated values of ALARM_8P with EV_ID1

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number (message 5) of MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Text number (message 3) of MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 6 | Subslot number | BYTE |

The process control messages of ALARM_8P with EV_ID2 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID2 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 00 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 01 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 02 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 03 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 04 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 05 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 06 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 07 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID2

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

Process control messages of ALARM_8P with EV_ID3 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID3 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 08 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 09 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | F |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 10 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 11 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 12 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 13 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 14 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 15 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID3

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

Process control messages for ALARM_8P with EV_ID4 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID_4 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 16 @4W%t# MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 17 @4W%t# MOD_D24_PN_TXT @ | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 18 @4W%t# MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 19 @4W%t# MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 20 @4W%t# MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 21 @4W%t# MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 22 @4W%t# MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 23 @4W%t# MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID4

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

For additional information, see chapter: Message Classes (Page 401).

## 3.45.5 MOD_D16_PN: Monitoring S7-300 SM modules with up to 16 channels and with diagnostics functions

### 3.45.5.1 Description of MOD_D16_PN

#### Object name (type + number)

FB 198

- I/Os of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN (Page 346)

#### Area of application

Block MOD_D16_PN monitors S7-300 SM modules with a maximum of 16 channels and with diagnostics functions.

### 3.45.5.2 I/Os of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CHAN_NUM | Number of channels | INT | 0 |
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number ALARM_8P_x (x = 1 - 2/1 - 3/1 - 4, assigned by the ES) | DWORD | 0 |
| FEATURE_xx | Feature parameter (xx = 01 - 04) | WORD | 0 |
| FEATURE_yy | Feature parameter (yy = 05 - 10) | DWORD | 0 |
| **LADDR** | Logic input address of the module | INT | 0 |
| **LADDR1** | Logical output address of the module (if output address is not the same as input address). | INT | 0 |
| MODE_xx | Operating mode channel xx (xx = 00 - 07/00 - 15/00 - 23) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| SUBSL_NO | Subslot number | BYTE | 0 |
| SUBN_TYP | 1 = external PN interface | BOOL | 0 |
| SUBN1_ID | PN IO system 1 ID (100 - 115) | BYTE | 255 |
| SUBN2_ID | PN IO system 2 ID (100 - 115) | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DIAG_INFx | System structure: Diagnostic information | STRUCT | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | 0 |
| EXT_STATx | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression active | DWORD | 16#00000000 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACKx | Message acknowledgment (x = 1 - 2/1 - 3/1 - 4) | WORD | 0 |
| MSGSTATx | Message error information (x = 1 - 2/1 - 3/1 - 4) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Operating mode channel xx (xx = 00 - 07/00 - 15/00 - 23) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN (Page 348)

Maintenance Status of MS (Page 403)

### 3.45.5.3 Message texts and associated values of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN

## Assignment of message text and message class

Process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | Module @1%d@/@2%d@/@3%d@/@6%d@: Removed | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@6%d@: Access error | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@6%d@: @5W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT@ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@6%d@: Multiple diagnostics interrupt | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@6%d@: @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@6%d@: @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@6%d@: @5W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT@ | F |
| | 8 | | No message |

Associated values of ALARM_8P with EV_ID1

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number (message 5) of MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Text number (message 3) of MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 6 | Subslot number | BYTE |

The process control messages of ALARM_8P with EV_ID2 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID2 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 00 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 01 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 02 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 03 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 04 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 05 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 06 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 07 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID2

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

Process control messages of ALARM_8P with EV_ID3 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID3 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 08 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 09 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | F |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 10 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 11 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 12 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 13 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 14 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 15 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID3

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

Process control messages for ALARM_8P with EV_ID4 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID_4 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 16 @4W%t# MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 17 @4W%t# MOD_D24_PN_TXT @ | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 18 @4W%t# MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 19 @4W%t# MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 20 @4W%t# MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 21 @4W%t# MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 22 @4W%t# MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 23 @4W%t# MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID4

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

For additional information, see chapter: Message Classes (Page 401).

## 3.45.6 MOD_D24_PN: Monitoring S7-300 SM modules with up to 24 channels and with diagnostics functions

### 3.45.6.1 Description of MOD_D24_PN

#### Object name (type + number)

FB 199

- I/Os of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN (Page 352)

#### Area of application

Block MOD_D24_PN monitors S7-300 SM modules with a maximum of 24 channels and with diagnostics functions.

### 3.45.6.2 I/Os of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CHAN_NUM | Number of channels | INT | 0 |
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number ALARM_8P_x<br>(x = 1 - 2/1 - 3/1 - 4, assigned by the ES) | DWORD | 0 |
| FEATURE_xx | Feature parameter (xx = 01 - 04) | WORD | 0 |
| FEATURE_yy | Feature parameter (yy = 05 - 10) | DWORD | 0 |
| **LADDR** | Logic input address of the module | INT | 0 |
| **LADDR1** | Logical output address of the module (if output address is not the same as input address). | INT | 0 |
| MODE_xx | Operating mode channel xx (xx = 00 - 07/00 - 15/00 - 23) | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| SUBSL_NO | Subslot number | BYTE | 0 |
| SUBN_TYP | 1 = external PN interface | BOOL | 0 |
| SUBN1_ID | PN IO system 1 ID (100 - 115) | BYTE | 255 |
| SUBN2_ID | PN IO system 2 ID (100 - 115) | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DIAG_INFx | System structure: Diagnostic information | STRUCT | 0 |
| DXCHG_xx | Bidirectional data exchange channel (xx = 00 - 15)<br>Bit = 0: Release for maintenance<br>Bit1 Byte0: Flutter suppression<br>Bit2 to Bit7 Byte0: Reserved<br>Byte 1: Reserved<br>Byte 2: Reserved<br>Byte 3: Fluttering time | DWORD | 0 |
| EXT_STATx | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression active | DWORD | 16#00000000 |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACKx | Message acknowledgment (x = 1 - 2/1 - 3/1 - 4) | WORD | 0 |
| MSGSTATx | Message error information (x = 1 - 2/1 - 3/1 - 4) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_xx | Operating mode channel xx (xx = 00 - 07/00 - 15/00 - 23) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN (Page 354)

Maintenance Status of MS (Page 403)

### 3.45.6.3 Message texts and associated values of MOD_D8_PN/MOD_D16_PN/MOD_D24_PN

## Assignment of message text and message class

Process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | Module @1%d@/@2%d@/@3%d@/@6%d@: Removed | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@6%d@: Access error | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@6%d@: @5W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT@ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@6%d@: Multiple diagnostics interrupt | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@6%d@: @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@6%d@: @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@6%d@: @5W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT@ | F |
| | 8 | | No message |

Associated values of ALARM_8P with EV_ID1

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number (message 5) of MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Text number (message 3) of MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 6 | Subslot number | BYTE |

The process control messages of ALARM_8P with EV_ID2 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID2 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 00 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 01 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 02 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 03 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 04 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 05 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 06 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 07 @4W%t# MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID2

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

Process control messages of ALARM_8P with EV_ID3 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID3 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 08 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 09 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | F |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 10 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 11 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 12 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 13 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 14 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 15 @4W%t# MOD_D16_PN_TXT /MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID3

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D8_PN_TXT /MOD_D16_PN_TXT /MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

Process control messages for ALARM_8P with EV_ID4 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID_4 | 1 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 16 @4W%t# MOD_D24_PN_TXT @ | S |
| | 2 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 17 @4W%t# MOD_D24_PN_TXT @ | S |
| | 3 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 18 @4W%t# MOD_D24_PN_TXT @ | S |
| | 4 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 19 @4W%t# MOD_D24_PN_TXT @ | S |
| | 5 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 20 @4W%t# MOD_D24_PN_TXT @ | S |
| | 6 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 21 @4W%t# MOD_D24_PN_TXT @ | S |
| | 7 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 22 @4W%t# MOD_D24_PN_TXT @ | S |
| | 8 | Module @1%d@/@2%d@/@3%d@/@5%d@: Error channel 23 @4W%t# MOD_D24_PN_TXT @ | S |

Associated values of ALARM_8P with EV_ID4

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Text number from MOD_D24_PN_TXT | BYTE |
| 5 | Subslot number | BYTE |

For additional information, see chapter: Message Classes (Page 401).

## 3.45.7 MOD_HA_PN: Monitoring Device-Specific Diagnostics of HART Field Devices

### 3.45.7.1 Description of MOD_HA_PN

#### Object name (type + number)

FB 200

- I/Os of MOD_HA_PN (Page 358)

#### Area of application

The MOD_HA_PN module reports diagnostic events of an HART field device that is connected to a channel of an ET 200M HART module. HART modules of ET 200iS are not supported. H systems support only the modules installed in switched racks.

### 3.45.7.2 I/Os of MOD_HA_PN

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Type | Default |
|---|---|---|---|
| **CHAN_NO** | Channel number | BYTE | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_IDx | Message number ALARM_8P_x (x = 1 - 2, assigned by the ES) | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| **MODE** | Channel operating mode | WORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | PN IO system 1 ID (100 - 115) | BYTE | 255 |
| **SUBN2_ID** | PN IO system 1 ID (100 - 115) | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Type | De-fault |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#0 0000 000 |
| DIAG_H | Diagnostic information of HART communication channel | STRUCT | |
| DXCHG | Bidirectional data exchange channel<br>Bit 0 = Release for maintenance<br>Bits 1-31 = Reserve | DWORD | 0 |
| EXT_STAT | Release for maintenance - extended status | DWORD | 0 |
| FS_ACTIVE | Flutter suppression | DWORD | 16#0 0000 000 |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSGSTAT | Status output of ALARM_8P_x (x = 1-2) | WORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE | Channel operating mode | DWORD | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QREC_ERR | 1 = Read diagnostic data error | BOOL | 0 |
| QREC_VAL | 1 = Read diagnostic data | BOOL | 0 |
| STATUS | Read diagnostics status | DWORD | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Type | De-fault |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | CPU diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_HA_PN (Page 360)

Maintenance Status of MS (Page 403)

### 3.45.7.3 Message texts and associated values of MOD_HA_PN

#### Assignment of message text and message class

Process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message block | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID1 (ALARM_8P) | 1 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Communication errors | S |
| | 2 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Error | S |
| | 3 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Secondary var. outside range | F |
| | 4 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Primary var. outside range | F |
| | 5 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Analog value specified | S |
| | 6 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Analog value saturated | S |
| | 7 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Maintenance alarm | S |
| | 8 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Further status available | F |

Associated values of ALARM_8P with EV_ID1

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Channel number | BYTE |

The process control messages of ALARM_8P with EV_ID2 are assigned as follows:

| Message block | Message no. | Default message text | Message class |
|---|---|---|---|
| EV_ID2 (ALARM_8P) | 1 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Reassignment of parameters | SA |
| | 2 | HART field device@1%d@/@2%d@/@3%d@/@4%d@: Cold restart | SA |
| | 3 | | No message |
| | 4 | | No message |
| | 7 | | No message |
| | 8 | | No message |

Associated values of ALARM_8P with EV_ID2

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the primary PN IO system | BYTE |
| 2 | Rack/station number | BYTE |
| 3 | Slot number | BYTE |
| 4 | Channel number | BYTE |

For additional information, see chapter: Message Classes (Page 401).

## 3.45.8 MOD_CP_PN: Monitoring of serial communication modules

### 3.45.8.1 Description of MOD_CP_PN

#### Object name (type + number)

FB 201

- I/Os of MOD_CP_PN (Page 362)

#### Area of application

The MOD_CP_PN block monitors a serial communication module CP 341 or CP 441.

### 3.45.8.2 I/Os of MOD_CP_PN

The factory setting of the block display in CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DELAY1 | Alarm delay 1 (s) | INT | 2 |
| DELAY2 | Alarm delay 2 (s) | INT | 2 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number ALARM_8P (assigned by the ES) | DWORD | 0 |
| **LADDR** | Logic address of the module | INT | 0 |
| MODE_00 | Channel 1 mode | WORD | 0 |
| MODE_01 | Channel 2 mode (CP 441 only) | WORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number | BYTE | 0 |
| **SUBSL_NO** | Subslot number | BYTE | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | PN IO system 1 ID (100 - 115) | BYTE | 255 |
| **SUBN2_ID** | PN IO system 2 ID (100 - 115) | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CH_ACTIVE | Channel active | DWORD | 16#00000000 |
| CH_EXIST | Channel exists | DWORD | 0 |
| CH_OK | Channel OK | DWORD | 0 |
| DIAG_INF | System structure: Diagnostic information | STRUCT | |
| MOD_INF | System structure: Module parameter | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| OMODE_00 | Value status/channel 1 mode | DWORD | 0 |
| OMODE_01 | Value status/channel 2 mode (CP 441 only) | DWORD | 0 |
| QERR | 1 = program error | BOOL | 1 |
| QMODF | 1 = module removed/defective | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = rack error | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_MODE | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | System structure: CPU diagnostics | STRUCT | |
| RAC_DIAG | System structure: Rack diagnostics | STRUCT | |

## Additional information

For additional information, refer to the following sections:

Message texts and associated values of MOD_CP_PN (Page 364)

Maintenance Status of MS (Page 403)

### 3.45.8.3 Message texts and associated values of MOD_CP_PN

## Assignment of message text and message class

Process control messages of ALARM_8P with EV_ID1 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | CP @1%d@/@2%d@/@3%d@: Removed | S |
| | 2 | CP @1%d@/@2%d@/@3%d@: Access error | S |
| | 3 | CP @1%d@/@2%d@/@3%d@: @4W%t#MOD_CP_TXT@ | S |
| | 4 | CP @1%d@/@2%d@/@3%d@: Wrong parameter | S |
| | 5 | CP @1%d@/@2%d@/@3%d@: Wire break | S |
| | 6 | CP @1%d@/@2%d@/@3%d@/2: Wrong parameter | S |
| | 7 | CP @1%d@/@2%d@/@3%d@/2: Wire break | S |
| | 8 | CP @1%d@/@2%d@/@3%d@: Multiple diagnostic interrupt | S |

Associated values of ALARM_8P with EV_ID1

| Associated value | Block parameter | Meaning |
|---|---|---|
| 1 | SUBN_ID | ID of the PN IO system (bytes) |
| 2 | RACK_NO | Rack/station number (byte) |
| 3 | SLOT_NO | Slot number (byte) |

The process control messages of ALARM_8P with EV_ID2 are assigned as follows:

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID2 | 1 | Module @1%d@: Uncertain, maintenance request | S |
| | 2 | Module @1%d@: Good, maintenance required | S |
| | 3 | | No message |
| | 4 | | No message |
| | 5 | | No message |
| | 6 | | No message |
| | 7 | | No message |
| | 8 | | No message |

Associated values of ALARM_8P with EV_ID2

| Associated value | Block parameter | Meaning |
|---|---|---|
| 1 | SUBN1_ID | ID of the primary PN IO system (bytes) |
| 2 | SUBN2_ID | ID of the redundant PN IO system (SUBN2_ID) (bytes) |
| 3 | SLOT_NO | Slot number (byte) |

For additional information, see chapter: Message Classes (Page 401).

## 3.45.9 PADP_L10_PN: Monitoring PA slaves downstream of DPV0 with up to 16 slots

### 3.45.9.1 Description of PADP_L10_PN

#### Object name (type + number)

FB 203

- I/Os of PADP_L10_PN (Page 366)

#### Area of application

Block PADP_L10_PN monitors DPV0 PA field devices with a maximum of 16 slots which are operated as DPV0 slaves on a DP master system, either directly or via a DP/PA coupler. The DP/PA coupler is connected downstream of an IE/PB link. The PA field devices must conform to the PROFIBUS V3.0 profile.

### 3.45.9.2 I/Os of PADP_L10_PN

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O | Meaning | Type | De-fault |
|---|---|---|---|
| **DADDR** | Diagnostic address of the PN IO device | INT | 0 |
| EN_DIAG | 1 = Queued diagnostic event | BOOL | 0 |
| MODE_xx | Mode slot (xx = 00 - 31) | WORD | 0 |
| **PADP_ADR** | Address of the PA field device | BYTE | 0 |
| PROF_V30 | 1 = PA slave profile V3.0 | BOOL | 0 |
| **RACK_NO** | Rack number | BYTE | 255 |
| **SLOT0_NO** | Slot number 0 of the slave at the IE/PB link | BYTE | 0 |
| **SUBN_TYP** | 1 = external PN interface | BOOL | 0 |
| **SUBN1_ID** | PN IO system 1 ID (100 - 115) | BYTE | 255 |
| **SUBN2_ID** | PN IO system 2 ID (100 - 115) | BYTE | 255 |

## Output parameters

| I/O | Meaning | Type | De-fault |
|---|---|---|---|
| DINFO | Diagnostic status of the PA field device | STRUCT | |
| EN_M_xx | Enable slot (xx = 0 - 31) | BOOL | 0 |
| OMODE_xx | Mode slot (xx = 16 - 31) | DWORD | 0 |
| PA_DIAG | PA field device diagnostic information | DWORD | 0 |
| QERR | 1 = program error | BOOL | 0 |
| QMODF | 1 = field device error/fault | BOOL | 0 |
| QPERAF | 1 = I/O access error | BOOL | 0 |
| QRACKF | 1 = Slave failure/faulty | BOOL | 0 |

## In-out parameters

| I/O | Meaning | Type | De-fault |
|---|---|---|---|
| CC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | System structure: CPU diagnostics | STRUCT | |
| CPU_OB_5X | OB_5x startup information | STRUCT | |
| RAC_DIAG | System structure: RACK diagnostics | STRUCT | |

## 3.45.10 OB_DIAG1_PN: OB diagnostics for avoiding stoppages in DPV1 master systems

### 3.45.10.1 Description of OB_DIAG1_PN

#### Object name (type + number)

FB 202

- I/Os of OB_DIAG1_PN (Page 368)

#### Area of application

Block OB_DIAG1_PN monitors the failure and recovery of DP or PA slaves. The slaves can be connected to an IE/PB. To prevent the CPU stopping, OB_DIAG1_PN blocks further evaluation if a slave is defective.

### 3.45.10.2 I/Os of OB_DIAG1_PN

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| **DADDR** | IE/PB link diagnostic address | INT | 0 |
| **DPA_LINK** | Slave connection: 0 = PN-IO Master 1 = IE/PB link | BOOL | 0 |
| EN_MSG | 1 = enable message | BOOL | 1 |
| EN_MSG_D | 1 = Enable message "Device failure" | BOOL | 1 |
| EV_ID | Message number ALARM_8P (assigned by the ES) | DWORD | 0 |
| **LADDR** | Logical basic address of the slave | INT | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **PADP_ADR** | Diagnostic address of the PN IO device | BYTE | 255 |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SLOT_NO** | Slot number of the slave at the IE/PB link | BYTE | 255 |
| **SUBN_TYP** | 1 = external PN interface | BOOL | 0 |
| **SUBN1_ID** | PN IO system 1 ID (100 - 115) | BYTE | 255 |
| **SUBN2_ID** | PN IO system 2 ID (100 - 115) | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_F | 1 = Enable function/function block | BOOL | 0 |
| MOD_INF | System structure: Module diagnostics | STRUCT | |
| MSG_ACK | Message acknowledgment | WORD | 0 |
| MSG_STATx | STATUS output of ALARM_8P_x (x = 1-2) | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| QRACKF | 1 = Slave failure/faulty | BOOL | 0 |
| RAC_DIAG | System structure: RACK diagnostics | STRUCT | |
| SUBN1ACT | 1 = PN IO device 1 active | BOOL | 0 |
| SUBN1ERR | 1 = Error in the PN IO device 1 | BOOL | 0 |
| SUBN2ACT | 1 = PN IO device 2 active | BOOL | 0 |
| SUBN2ERR | 1 = Error in the PN IO device 2 | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG_PN | System structure: CPU diagnostics | STRUCT | |
| CPU_OB_5X | OB_5x start information | STRUCT | |
| RAC_DIAG_I | System structure: RACK diagnostics | STRUCT | |
| SUB_DIAG | OB startup information | STRUCT | |

### 3.45.10.3 Message texts and associated values of OB_DIAG1_PN

#### Assignment of message text and message class

| Message block ALARM_8P | Message number | Default message text | Message class |
|---|---|---|---|
| EV_ID1 | 1 | PN IO device @1%d@/ @2%d@: Redundancy loss | F |
| | 2 | PN IO device @1%d@/ @2%d@: Failure | S |
| | 3 | PN IO device @1%d@/ @2%d@/@3%d@ : Multiple failure | S |
| | 4 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 82) | S |
| | 5 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 55) | S |
| | 6 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 56) | S |
| | 7 | Device @1%d@/ @2%d@/@3%d@: Multiple alarm (OB 57) | S |
| | 8 | Device @1%d@/ @2%d@/@3%d@: Failure | S |

#### Assignment of Associated Values

| Associated value | Block parameter | Data type |
|---|---|---|
| 1 | ID of the primary PN IO system (SUBN1_ID) | BYTE |
| 2 | Rack/station number (RACK_NO) | BYTE |
| 3 | Slot number (SLOT_NO) | BYTE |

## 3.45.11 DPAY_V1_PN: Enabling blocks downstream of a DP/PA and Y-link operating as a V1 Slave

### 3.45.11.1 Description of DPAY_V1_PN

#### Object name (type + number)

FB 204

- I/Os of DPAY_V1_PN (Page 371)

#### Area of application

The DPAY_V1_PN block releases the field device-specific blocks downstream of the IE/PB link.

The IE/PB link operates as a PA master for the lower-level PA field devices and as a slave on the IE bus.

#### See also

General Information About Block Description (Page 9)

### 3.45.11.2 I/Os of DPAY_V1_PN

The factory setting of the block display in CFC is identified in the "I/O" column:

I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| DPPA_00 | Information about the DP/PA slave (xx = 00 - 63) | STRUCT | |
| **RACK_NO** | Rack number | BYTE | 0 |
| **SUBN1_ID** | PN IO system 1 ID (100 - 115) | BYTE | 255 |
| **SUBN2_ID** | PN IO system 2 ID (100 - 115) | BYTE | 255 |

## Output parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| EN_Mxx | 1 = Enable slave (xx = 00 - 63) | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | Default |
|---|---|---|---|
| CPU_DIAG_PN | System structure: CPU diagnostics | STRUCT | |
| CPU_OB_5X | OB_5x startup information | STRUCT | |

# 3.46 SUBNET: DP master system monitoring

## 3.46.1 Description of SUBNET

### Object name (type + number)

FB 106

- SUBNET block I/Os (Page 377)

### Area of application

The SUBNET block is used to shorten acyclic OB processing times. Only the blocks that are actually affected can be called in the case of an acyclic event.

### Calling OBs

The SUBNET block must be installed in the run sequence in the following OBs:

| | |
|---|---|
| OB 1 | Cyclic program |
| OB 55 | Status interrupt (only if a DP/PA slave is required) |
| OB 56 | Update interrupt (only if a DP/PA slave is required) |
| OB 57 | Vendor-specific alarm (only if a DP/PA slave is required) |
| OB 70 | I/O redundancy error |
| OB 72 | CPU redundancy error |
| OB 81 | Power supply error |
| OB 82 | Diagnostic interrupt |
| OB 83 | Insert/remove module interrupt |
| OB 85 | Program execution error |
| OB 86 | Rack failure |
| OB 100 | Warm restart |

## Use in CFC

The following actions are executed automatically with the "**Generate module drivers**" CFC function:

- Runtime groups with driver/system blocks are created and organized by rack.
- The SUBN1_ID, SUBN2_ID and SUBN_TYP inputs are configured.
- The EN_Rxxx outputs are interconnected with the relevant blocks (e.g., RACK).
- The IN_OUT structure CPU_DIAG is interconnected with the OUT structure of the OB_BEGIN block.
- The IN_OUT structure SZL_71 is interconnected with the OUT structure of the OB_BEGIN block.
- The OUT structure SUB_DIAG is interconnected with the IN_OUT structures of the affected blocks (such as RACK).

## Function and method of operation

The SUBNET block monitors a DP master system, and enables the blocks (such as RACK) for processing the connected DP slaves (such as ET 200M). Corresponding messages are generated, and the SUBN1ERR and SUBN2ERR outputs are set if a DP master system fails or loses redundancy. The SUB_DIAG output structure contains the geographic address of the DP Master system (and of DP Master system 2 in H systems), as well as the group error information SUBN0_ERR (for DP master system 1) and SUBN1_ERR (for DP master system 2). If SUBN0_ERR = 1 or SUBN1_ERR = 1, the corresponding DP master system is **not** available.

The SUBNET block is installed in each connected DP master system or in the OBs listed above once for the local I/O devices. It is enabled by OB_BEGIN. Start and diagnostic information is read from the CPU_DIAG structure. It is interconnected with the CPU_DIAG structure of the OB_BEGIN. The SUBNET block is assigned one enable output for each connectable rack or DP Master system (for each expansion rack for central I/Os). It uses the start information of the calling OB to determine whether the reported event occurred at its DP master (or at the central I/O), and then sets the output for the affected rack or DP master system (EN_Rxxx).

If redundant DP master systems are used (for H CPUs only), a rack (such as ET 200M) is connected to the two DP masters, and is assigned the same station number at both. The SUBNET block has two input parameters (SUBNx_ID), and the type identifier SUBN_TYP for this function. If the integrated interface of the CPU module is the DP master, SUBN_TYP = FALSE, otherwise SUBN_TYP = TRUE.

The MASTER_0 and MASTER_1 outputs indicate which CPU is currently the master.

If a DP master fails, the system sets all EN_Rxxx = TRUE, and reports a redundancy loss or failure. The return of redundancy or the DP master is reported when a failed DP slave has reestablished the connection.

The status of the DP master system, the set SUBNx_ID and type identifier are saved in the output structure SUB_DIAG.

If a "power supply error" (OB 81) event occurs, the SUBNET block will enable only those RACK blocks that are expansion racks, which is indicated by SUBNx_ID = 0.

Note: If you want to change the SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs online, you must set input ACC_ID = TRUE. to update the output values.

## Redundancy

The SUBNET block supports redundancy of DP master systems of the 414-H/417-H CPU if distributed I/Os are used. To use this function, you must configure the SUBN1_ID (connection to CPU 0) and SUBN2_ID (connection to CPU 1) inputs with the numbers of the redundant DP master systems. If there is no redundancy, the remaining input must be assigned the value 16#FF (default).

## Error handling

Error handling for the block is limited to the evaluation of the error information of ALARM_8P. You will find additional information about error handling in "Error information of output parameter MSG_STAT (Page 398)".

## Startup/initial startup behavior

The SUBNET block initializes the messages of ALARM_8P.

The operating mode of the DP Master system is checked, and entered in the SUB_DIAG.V1_MODE structure with SSL 0X90H (0 = compatibility mode, 1 = DPV1 mode). If DPV1 mode is active, the CPU_DIAG.MODE_V1 structure is also set to TRUE.

## Overload behavior

The SUBNET block counts the OB 86 calls (failures only). The counter is reset in OB1. If more than two OB 86 failure events occur in succession before the cycle control point (OB1) is reached, these are rejected and a message "Failure OB 86 DP master system:x" is output. If an OB 86 call is rejected, the DP master system is registered as having failed.

## Time response

Not available

## Message response

After being called by an OB 86, OB 70 and OB 72, the block analyzes the status of its assigned DP master system, and generates the relevant messages for redundancy loss or DP master system failure by broadcasting an ALARM_8P.
The message function can be disabled by setting EN_MSG = FALSE.

The SUBNET block generally reports only events triggered in the DP master system it monitors.

Exception: If a CPU fails in the H system, the following messages are generated:

- in a non-redundant DP master system: “DP master failure” message
- in a redundant DP master system: “DP master redundancy loss” message

## Operator control and monitoring

Note: If you selected the "OCM possible" option in the block object properties in the CFC, the variables transferred to the OS are identified under "I/Os of ..." (OCM column, "+"). Default: Option not activated.

If asset management is used in the project and the diagnostic screens have been generated, the faceplate can be called via its block icon.

For more information, refer to the "Process Control System PCS 7; Maintenance Station" manual.

## Additional information

For additional information, refer to the following sections:

Message Texts and associated values of SUBNET (Page 379)

Maintenance status of MS (Page 403)

### 3.46.2 I/Os of SUBNET

The factory setting of the block display in the CFC is identified in the "I/O" column:
I/O name **bold** = I/O visible, I/O name normal = I/O not visible.

You will find explanations of and information on abbreviations in the section: "General Information About Block Description (Page 9)".

#### Input parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| EN_MSG | 1 = enable message | BOOL | 1 |
| EV_ID | Message number | DWORD | 0 |
| **MS** | Maintenance status | DWORD | 0 |
| **SUBN_TYP** | 1 = external DP interface | BOOL | 0 |
| **SUBN1_ID** | ID of the primary DP master system | BYTE | 255 |
| **SUBN2_ID** | ID of the redundant DP master system | BYTE | 255 |

#### Output parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| EN_Rxxx | 1 = Enable rack (xxx = 0 - 127) | BOOL | 0 |
| **MASTER_0** | 1 = Master CPU in rack 0 | BOOL | 0 |
| **MASTER_1** | 1 = Master CPU in rack 1 | BOOL | 0 |
| MSG_STAT | Message error information | WORD | 0 |
| O_MS | Maintenance status | DWORD | 0 |
| **SUB_DIAG** | System structure: CPU diagnostics | STRUCT | |
| **SUBN1ERR** | 1 = Error in DP master system 1 | BOOL | 0 |
| **SUBN2ERR** | 1 = Error in DP master system 2 | BOOL | 0 |

## In-out parameters

| I/O (parameter) | Meaning | Data type | De-fault |
|---|---|---|---|
| ACC_ID | 1 = accept MODE settings | BOOL | 0 |
| CPU_DIAG | CPU diagnostics | STRUCT | |
| CPU_OB_5X | OB_5x startup information | STRUCT | |
| SZL_71 | System structure SZL71 | STRUCT | |

**Note**

The maximum number of racks is determined by the address volume of PROFIBUS. All available CPUs can thus be used. The entire address volume is used by the CPU 417-4.

## Additional information

For additional information, refer to the following sections:

Message Texts and associated values of SUBNET (Page 379)

Maintenance status of MS (Page 403)

### 3.46.3 Message texts and associated values of SUBNET

#### Assignment of message text and message class

| Message no. | Default message text | Message class |
|---|---|---|
| 1 | DP master @1%d@: Redundancy loss | F |
| 2 | DP master @2%d@: Redundancy loss | F |
| 3 | DP master @1%d@: Failure | S |
| 4 | DP master @2%d@: Failure | S |
| 5 | DP master @2%d@: Multiple failure | S |
| 6 | CPU loss of redundancy in rack @4%d@ | F |

#### Assignment of associated values

| Associated value | Block parameters |
|---|---|
| 1 | ID of the primary DP master system (SUBN1_ID) |
| 2 | ID of the redundant DP master system (SUBN2_ID) |
| 3 | Multiple failure, ID of DP master system |

#### See also

Message Classes (Page 401)

# 4 Internal block

## 4.1 ChkREAL: Internal Block

### Object name (type + number)

FC260

This block is a system block and is only used internally. There is therefore no help available for it.

## 4.2 QC_CHNG: Internal block

### Object name (type + number)

FB 135

This block is a system block and is only used internally. There is therefore no help available for it.

# Appendix 5

## 5.1 "Blocks - basic library" technical data

### Overview

The table below contains the technical data for the blocks. The table columns have the following meanings:

- Block type name

  The symbolic identifier in the symbol table of the library for the relevant FB or FC. Must be unique within the project.
- Object name

  Consists of the block type (FB, FC) and number.
- Typical execution time

  CPU runtime for processing the corresponding block program under normal circumstances (for example, for a driver, this is the execution time in the cyclic interrupt OB (OB3x), without generation of a channel error message).

  The table below shows the runtime of blocks in a 417-4 CPU. The block runtime on other CPUs depends on the CPU performance.
- Block length in load/work memory

  Memory requirements of the program code, once for each block type.
- Length of instance data in load/work memory

  Memory requirement of an instance DB.
- Temporary memory

  The local-data memory required in a priority class when the block is called. This limit is CPU specific. When it is exceeded, you have to check the CPU configuration in the local-data memory and, if necessary, distribute it amongst the OBs to meet the actual requirements.
- Multiple instance block

  The specified blocks are used by the block concerned, and must exist in the user program. They can be found in the same library.

| Block<br>(Type name) | FB/FC no. | Typical run time CPU 417-4 (µs) | Block length in load/work memory (bytes) | Length of the instance data in the load/work memory (bytes) | Temporary memory (bytes) | Multiple instance block |
|---|---|---|---|---|---|---|
| ChkREAL | FC 260 | | | | | |
| CONEC | FB 88 | 98 | 10266 / 8642 | 1186 / 340 | 84 | 16 x SFB 35 |
| CPU_RT | FB 128 | 67 | 31434 / 27370 | 2800 / 1784 | 86 | |
| DIAG_AB | FB 414 | | | | | |
| DPAY_V0 | FB 108 | 159 | 1792 / 1202 | 542 / 70 | 22 | |
| DPAY_V1 | FB 115 | 155 | 11206 / 8506 | 3588 / 1388 | 136 | SFB 35 |
| DPDIAGV0 | FB 117 | 115 | 3980 / 2184 | 1800 / 194 | 66 | |
| DREP | FB 113 | 19 | 4202 / 3038 | 1296 / 366 | 124 | |
| DREP_L | FB 125 | 20 | 6578 / 5358 | 1406 / 486 | 52 | 2 x SFB 35 |
| FM_CNT | FB 126 | 36 | 1496 / 1060 | 546 / 140 | 10 | |
| FM_CO | FB 79 | 18 | 3132 / 1780 | 1732 / 566 | 46 | |
| IMDRV_TS | FB 129 | | | | | |
| MOD_1 | FB 91 | 91 | 13202/ 10498 | 5946 / 4060 | 116 | 16 x SFB 35 |
| MOD_2 | FB 92 | 91 | 4912 / 3862 | 1120 / 346 | 68 | SFB 35 |
| MOD_3 | FB 95 | 91 | 4984 / 3868 | 1280 / 442 | 66 | SFB 35 |
| MOD_4 | FB 119 | 16 | 4988 / 3872 | 1288 / 448 | 66 | SFB 35 |
| MOD_64 | FB 137 | | | | | SFB 35 |
| MOD_CP | FB 98 | 104 | 3496 / 2540 | 1108 / 346 | 52 | SFB 35 |
| MOD_D1 | FB 93 | 96 | 6850 / 5622 | 1186 / 340 | 80 | SFB 35 |
| MOD_D2 | FB 94 | 97 | 12552 / 10752 | 1818 / 700 | 86 | 3 x SFB 35 |
| MOD_D3 | FB 134 | 103 | 13432 / 11442 | 3958 / 2164 | 90 | 3 x SFB 35 |
| MOD_HA | FB 97 | 18 | 10836 / 8938 | 2440 / 1090 | 82 | 5 x SFB 35 |
| MOD_MS | FB 96 | 99 | 5442 / 4282 | 1356 / 464 | 54 | SFB 35 |
| MOD_PAL0 | FB 99 | 169 | 7758 / 6322 | 1814 / 740 | 84 | 2 x SFB 35 |
| MOD_PAX0 | FB 112 | 112 | 4470 / 3746 | 1006 / 490 | 50 | 2 x SFB 35<br>SFB 52 |
| MODB_341 | FB 80 | 594 | 4388 / 3666 | 1012 / 490 | 54 | 2 x SFB 35 |
| OB_BEGIN | FB 100 | 158 | 3012 / 2268 | 1206 / 630 | 120 | |
| OB_DIAG1 | FB 118 | 23 | 10886 / 8924 | 1690 / 306 | 116 | SFB 35<br>SFB 52 |
| OB_END | FC 280 | 4 | 514 / 86 | - / - | 4 | |
| OR_HA16C | FB 133 | 181 | 8492 / 6972 | 2362 / 1146 | 70 | 5 x SFB 35 |
| OR_M_16 | FB 81 | 181 | 3682 / 2736 | 1176 / 410 | 50 | SFB 35 |
| OR_M_16C | FB 84 | 183 | 8010 / 6516 | 2356 / 1146 | 70 | SFB 35 |
| OR_M_32 | FB 82 | 268 | 3778 / 2736 | 1464 / 602 | 50 | SFB 35 |
| OR_M_32C | FB 85 | 374 | 12618 / 10436 | 3958 / 2164 | 70 | 9 x SFB 35 |
| OR_M_8C | FB 83 | 94 | 5926 / 4730 | 1656 / 698 | 70 | |
| PADP_L00 | FB 109 | 15 | 3526 / 2690 | 904 / 262 | 40 | SFB 35 |

| Block (Type name) | FB/FC no. | Typical run time CPU 417-4 (µs) | Block length in load/work memory (bytes) | Length of the instance data in the load/work memory (bytes) | Temporary memory (bytes) | Multiple instance block |
|---|---|---|---|---|---|---|
| PADP_L01 | FB 110 | 19 | 4642 / 3600 | 1410 / 578 | 40 | 3 x SFB 35 |
| PADP_L02 | FB 111 | 23 | 6170 / 4890 | 201 / 954 | 40 | 5 x SFB 35 |
| PADP_L10 | FB 116 | 80 | 4998 / 3516 | 1460 / 228 | 56 | SFB 52 |
| PO_UPDAT | FC 279 | | 328 / 256 | - / - | 10 | |
| PS | FB 89 | 12 | 3062 / 2226 | 816 / 196 | 74 | |
| QC_CHNG | FB 135 | | | | | |
| RACK | FB 107 | 102 | 822 / 7484 | 1102 / 248 | 102 | SFB 35 |
| REC_BO | FB 208 | 69 | 3246 / 2356 | 992 / 128 | 2 | SFB 13 |
| REC_R | FB 210 | 69 | 1838 / 1332 | 956 / 476 | 2 | SFB 13 |
| RED_F | FC 289 | 41 | 5234 / 5020 | - / - | 24 | |
| SEND_BO | FB 207 | 163 | 2298 / 1668 | 718 / 110 | 2 | SFB 12 |
| SEND_R | FB 209 | 195 | 4486 / 3886 | 908 / 478 | 2 | SFB 12 |
| SUBNET | FB 106 | 308 | 6800 / 4920 | 1736 / 234 | 112 | SFB 35 |

## 5.2 MODE settings for FF devices

MODE_xx input parameters are available for up to 32 values of a FF field device.

| Block | I/O (parameters) (cyclic data) permitted combinations and sequence | Input (I)/Output (O) (PLS view) | MODE 16#xxyy O=xx, I=yy |
|---|---|---|---|
| Analog input (FbAnIn) | OUT | I | 16#0001 |
| Analog output (FbAnOu) | SP | O | 16#0100 |
| Discrete input (FbDiIn) | OUT_D | I | 16#0002 |
| Discrete output (FbDiOu) | SP_D | O | 16#0400 |

# 5.3 MODE settings for SM modules

## Measuring range coding of the analog input modules

Depending on the measuring-range coding of the analog input modules, the parameter MODE_xx (measuring-range coding) corresponding to the channel must be specified in accordance with the table. When thermocouples are used there are various options for combining the measuring type (coding A) with the measuring range (coding B). In this case, the MODE_xx parameter must be calculated according to the following formula and the result written to the MODE input as an INTEGER value:
Measuring range coding = 256 * coding A + coding B

Please note: The table shows codes **A** and **B** in binary format, and as the result the measuring range coding as a hexadecimal number.

| Measuring type | Coding (A) | Measuring range | Code (B) | MODE (256*A+B) |
|---|---|---|---|---|
| Deactivated | | | | 16#0000 |
| Voltage | 2#0001 | ± 25 mV | 2#1010 | 16#010A |
| | | ± 50 mV | 2#1011 | 16#010B |
| | | ± 80 mV | 2#0001 | 16#0101 |
| | | ± 250 mV | 2#0010 | 16#0102 |
| | | ± 500 mV | 2#0011 | 16#0103 |
| | | ± 1 V | 2#0100 | 16#0104 |
| | | ± 2.5 V | 2#0101 | 16#0105 |
| | | ± 5 V | 2#0110 | 16#0106 |
| | | 1 V to 5 V | 2#0111 | 16#0107 |
| | | 0 to 10 V | 2#1000 | 16#0108 |
| | | ± 10 V | 2#1001 | 16#0109 |
| | | ± 100 mV | 2#1100 | 16#010C |
| 4-wire measuring transducer | 2#0010 | ±3.2 mA | 2#0000 | 16#0200 |
| | | ± 5 mA | 2#0101 | 16#0205 |
| | | ± 10 mA | 2#0001 | 16#0201 |
| | | 0 mA to 20 mA | 2#0010 | 16#0202 |
| | | 4 mA to 20 mA | 2#0011 | 16#0203 |
| | | ± 20 mA | 2#0100 | 16#0204 |
| HART interface | 2#0111 | 4 to 20 mA (variant 0) OMODE settings for SM modules (Page 396) | 2#1100 | 16#070C |
| 2-wire measuring transducer | 2#0011 | 0 mA to 20 mA[1] | 2#0010 | 16#0302 |
| | | 4 to 20 mA | 2#0011 | 16#0303 |
| | | ± 20 mA | 2#0100 | 16#0304 |

| Measuring type | Coding (A) | Measuring range | Code (B) | MODE (256*A+B) |
|---|---|---|---|---|
| Resistor 4-wire connection | 2#0100 | 48 Ω | 2#0000 | 16#0400 |
| | | 150 Ω | 2#0010 | 16#0402 |
| | | 300 Ω | 2#0100 | 16#0404 |
| | | 600 Ω | 2#0110 | 16#0406 |
| | | 1000 Ω | 2#0111 | 16#040E |
| | | 3000 Ω | 2#0111 | 16#0407 |
| | | 6000 Ω | 2#1000 | 16#0408 |
| | | PTC | 2#1111 | 16#040F |
| Resistor 3-wire connection | 2#0101 | 48 Ω | 2#0000 | 16#0500 |
| | | 150 Ω | 2#0010 | 16#0502 |
| | | 300 Ω | 2#0100 | 16#0504 |
| | | 600 Ω | 2#0110 | 16#0506 |
| | | 1000 Ω | 2#0111 | 16#050E |
| | | 3000 Ω | 2#0111 | 16#0507 |
| | | 6000 Ω | 2#1000 | 16#0508 |
| | | PTC | 2#1111 | 16#050F |
| Resistor 2-wire connection | 2#0110 | 48 Ω | 2#0000 | 16#0600 |
| | | 150 Ω | 2#0010 | 16#0602 |
| | | 300 Ω | 2#0100 | 16#0604 |
| | | 600 Ω | 2#0110 | 16#0606 |
| | | 1000 Ω | 2#0111 | 16#060E |
| | | 3000 Ω | 2#0111 | 16#0607 |
| | | 6000 Ω | 2#1000 | 16#0608 |
| | | PTC | 2#1111 | 16#060F |
| Thermocouple + linear, 4-wire connection | 2#1000 | Pt 100 climate range | 2#0000 | 16#0800 |
| | | Pt 200 climate range | 2#0111 | 16#0807 |
| | | Pt 500 climate range | 2#1000 | 16#0808 |
| | | Pt 1000 climate range | 2#1001 | 16#0809 |
| | | Ni 100 climate range | 2#0001 | 16#0801 |
| | | Ni 1000/LG-Ni 1000 climatic range | 2#1010 | 16#080A |
| | | Pt 100 standard range | 2#0010 | 16#0802 |
| | | Pt 200 standard range | 2#0011 | 16#0803 |
| | | Pt 500 standard range | 2#0100 | 16#0804 |
| | | Pt 1000 standard range | 2#0101 | 16#0805 |
| | | Ni 100 standard range | 2#1011 | 16#080B |
| | | Ni 1000/LG-Ni 1000 standard range | 2#0110 | 16#0806 |
| | | Ni 120 standard range | 2#1100 | 16#080C |
| | | Ni 120 climate range | 2#1101 | 16#080D |
| | | Cu 10 climate range | 2#1110 | 16#080E |

| Measuring type | Coding (A) | Measuring range | Code (B) | MODE (256*A+B) |
|---|---|---|---|---|
| | | Cu 10 standard range | 2#1111 | 16#080F |
| | | Ni 200 standard range | 2#10000 | 16#0810 |
| | | Ni 200 climate range | 2#10001 | 16#0811 |
| | | Ni 500 standard range | 2#10010 | 16#0812 |
| | | Ni 500 climate range | 2#10011 | 16#0813 |
| | | Pt 10 GOST climatic | 2#10100 | 16#0814 |
| | | Pt 10 GOST standard (TC = 3910) | 2#10101 | 16#0815 |
| | | Pt 50 GOST climatic | 2#10110 | 16#0816 |
| | | Pt 50 GOST standard (TC = 3910) | 2#10111 | 16#0817 |
| | | Pt 100 GOST climatic | 2#11000 | 16#0818 |
| | | Pt 100 GOST standard (TC = 3910) | 2#11001 | 16#0819 |
| | | Pt 500 GOST climatic | 2#11010 | 16#081A |
| | | Pt 500 GOST standard (TC = 3910) | 2#11011 | 16#081B |
| | | Cu 10 GOST climatic | 2#11100 | 16#081C |
| | | Cu 10 GOST standard (TC = 426) | 2#11101 | 16#081D |
| | | Cu 50 GOST climatic | 2#11110 | 16#081E |
| | | Cu 50 GOST standard (TC = 426) | 2#11111 | 16#081F |
| | | Cu 100 GOST climatic | 2#100000 | 16#0820 |
| | | Cu 100 GOST standard (TC = 426) | 2#100001 | 16#0821 |
| | | Ni 100 GOST climatic | 2#100010 | 16#0822 |
| | | Ni 100 GOST standard | 2#100011 | 16#0823 |
| | | Pt 10 GOST standard (TC = 3850) | 2#1010101 | 16#0855 |
| | | Pt 50 GOST standard (TC = 3850) | 2#1010111 | 16#0857 |
| | | Pt 100 GOST standard (TC = 3850) | 2#1011001 | 16#0859 |
| | | Pt 500 GOST standard (TC = 3850) | 2#1011011 | 16#085B |
| | | Cu 10 GOST standard (TC = 428) | 2#10011101 | 16#089D |
| | | Cu 50 GOST standard (TC = 428) | 2#10011111 | 16#089F |
| | | Cu 100 GOST standard (TC = 428) | 2#10100001 | 16#08A1 |
| | | | | |

| Measuring type | Coding (A) | Measuring range | Code (B) | MODE (256*A+B) |
|---|---|---|---|---|
| Thermocouple + linear, 3-wire connection | 2#1001 | Pt 100 climate range | 2#0000 | 16#0900 |
| | | Pt 200 climate range | 2#0111 | 16#0907 |
| | | Pt 500 climate range | 2#1000 | 16#0908 |
| | | Pt 1000 climate range | 2#1001 | 16#0909 |
| | | Ni 100 climate range | 2#0001 | 16#0901 |
| | | Ni 1000/LG-Ni 1000 climatic range | 2#1010 | 16#090A |
| | | Pt 100 standard range | 2#0010 | 16#0902 |
| | | Pt 200 standard range | 2#0011 | 16#0903 |
| | | Pt 500 standard range | 2#0100 | 16#0904 |
| | | Pt 1000 standard range | 2#0101 | 16#0905 |
| | | Ni 100 standard range | 2#1011 | 16#090B |
| | | Ni 1000/LG-Ni 1000 standard range | 2#0110 | 16#0906 |
| | | Ni 120 standard range (variant 0)<br>KTY83/110 (variant 1) OMODE settings for SM modules (Page 396) | 2#1100 | 16#090C |
| | | Ni 120 climate range (variant 0)<br>KTY84/130 (variant 1) OMODE settings for SM modules (Page 396) | 2#1101 | 16#090D |
| | | Cu10 climate range | 2#1110 | 16#090E |
| | | Cu10 standard range | 2#1111 | 16#090F |
| | | Ni 200 standard range | 2#10000 | 16#0910 |
| | | Ni 200 climate range | 2#10001 | 16#0911 |
| | | Ni 500 standard range | 2#10010 | 16#0912 |
| | | Ni 500 climate range | 2#10011 | 16#0913 |
| | | Pt 10 GOST climatic | 2#10100 | 16#0914 |
| | | Pt 10 GOST standard (TC = 3910) | 2#10101 | 16#0915 |
| | | Pt 50 GOST climatic | 2#10110 | 16#0916 |
| | | Pt 50 GOST standard (TC = 3910) | 2#10111 | 16#0917 |
| | | Pt 100 GOST climatic | 2#11000 | 16#0918 |
| | | Pt 100 GOST standard (TC = 3910) | 2#11001 | 16#0919 |
| | | Pt 500 GOST climatic | 2#11010 | 16#091A |
| | | Pt 500 GOST standard (TC = 3910) | 2#11011 | 16#091B |
| | | Cu 10 GOST climatic | 2#11100 | 16#091C |

| Measuring type | Coding (A) | Measuring range | Code (B) | MODE (256*A+B) |
|---|---|---|---|---|
| | | Cu 10 GOST standard (TC = 426) | 2#11101 | 16#091D |
| | | Cu 50 GOST climatic | 2#11110 | 16#091E |
| | | Cu 50 GOST standard (TC = 426) | 2#11111 | 16#091F |
| | | Cu 100 GOST climatic | 2#100000 | 16#0920 |
| | | Cu 100 GOST standard (TC = 426) | 2#100001 | 16#0921 |
| | | Ni 100 GOST climatic | 2#100010 | 16#0922 |
| | | Ni 100 GOST standard | 2#100011 | 16#0923 |
| | | Pt 10 GOST standard (TC = 3850) | 2#1010101 | 16#0955 |
| | | Pt 50 GOST standard (TC = 3850) | 2#1010111 | 16#0957 |
| | | Pt 100 GOST standard (TC = 3850) | 2#1011001 | 16#0959 |
| | | Pt 500 GOST standard (TC = 3850) | 2#1011011 | 16#095B |
| | | Cu 10 GOST standard (TC = 428) | 2#10011101 | 16#099D |
| | | Cu 50 GOST standard (TC = 428) | 2#10011111 | 16#099F |
| | | Cu 100 GOST standard (TC = 428) | 2#10100001 | 16#09A1 |
| Thermocouple + linear, 2-wire connection | 2#1111 | Pt 100 climate range | 2#0000 | 16#0F00 |
| | | Pt 200 climate range | 2#0111 | 16#0F07 |
| | | Pt 500 climate range | 2#1000 | 16#0F08 |
| | | Pt 1000 climate range | 2#1001 | 16#0F09 |
| | | Ni 100 climate range | 2#0001 | 16#0F01 |
| | | Ni 1000/LG-Ni 1000 climatic range | 2#1010 | 16#0F0A |
| | | Pt 100 standard range | 2#0010 | 16#0F02 |
| | | Pt 200 standard range | 2#0011 | 16#0F03 |
| | | Pt 500 standard range | 2#0100 | 16#0F04 |
| | | Pt 1000 standard range | 2#0101 | 16#0F05 |
| | | Ni 100 standard range | 2#1011 | 16#0F0B |
| | | Ni 1000/LG-Ni 1000 standard range | 2#0110 | 16#0F06 |
| | | Ni 120 standard range | 2#1100 | 16#0F0C |
| | | Ni 120 climate range | 2#1101 | 16#0F0D |
| | | Cu10 climate range | 2#1110 | 16#0F0E |
| | | Cu10 standard range | 2#1111 | 16#0F0F |
| | | Ni 200 standard range | 2#10000 | 16#0F10 |

| Measuring type | Coding (A) | Measuring range | Code (B) | MODE (256*A+B) |
|---|---|---|---|---|
| | | Ni 200 climate range | 2#10001 | 16#0F11 |
| | | Ni 500 standard range | 2#10010 | 16#0F12 |
| | | Ni 500 climate range | 2#10011 | 16#0F13 |
| Thermocouple, linear, reference temperature 0 ºC / no reference point | 2#1010 | Type B [PtRh-PtRh] | 2#0000 | 16#0A00 |
| | | Type N [NiCrSi-NiSi] | 2#0001 | 16#0A01 |
| | | Type E [NiCr-CuNi] | 2#0010 | 16#0A02 |
| | | Type R [PtRh-Pt] | 2#0011 | 16#0A03 |
| | | Type S [PtRh-Pt] | 2#0100 | 16#0A04 |
| | | Type J [Fe-CuNi IEC] | 2#0101 | 16#0A05 |
| | | Type L [Fe-CuNi DIN] | 2#0110 | 16#0A06 |
| | | Type T [Cu-CuNi IEC] | 2#0111 | 16#0A07 |
| | | Type K [NiCr-Ni] | 2#1000 | 16#0A08 |
| | | Type U [Cu-CuNi DIN] | 2#1001 | 16#0A09 |
| | | Type C | 2#1010 | 16#0A0A |
| | | Type TXK/XK(L) | 2#1011 | 16#0A0B |
| Thermocouple, linear, reference temperature 50 ºC | 2#1011 | Type B [PtRh-PtRh] | 2#0000 | 16#0B00 |
| | | Type N [NiCrSi-NiSi] | 2#0001 | 16#0B01 |
| | | Type E [NiCr-CuNi] | 2#0010 | 16#0B02 |
| | | Type R [PtRh-Pt] | 2#0011 | 16#0B03 |
| | | Type S [PtRh-Pt] | 2#0100 | 16#0B04 |
| | | Type J [Fe-CuNi IEC] | 2#0101 | 16#0B05 |
| | | Type L [Fe-CuNi DIN] | 2#0110 | 16#0B06 |
| | | Type T [Cu-CuNi IEC] | 2#0111 | 16#0B07 |
| | | Type K [NiCr-Ni] | 2#1000 | 16#0B08 |
| | | Type U [Cu-CuNi DIN] | 2#1001 | 16#0B09 |
| | | Type C | 2#1010 | 16#0B0A |
| | | Type TXK/XK(L) | 2#1011 | 16#0B0B |
| Themocouple, fix. ref. temp | 2#1100 | Type B [PtRh-PtRh] | 2#0000 | 16#0C00 |
| | | Type N [NiCrSi-NiSi] | 2#0001 | 16#0C01 |
| | | Type E [NiCr-CuNi] | 2#0010 | 16#0C02 |
| | | Type R [PtRh-Pt] | 2#0011 | 16#0C03 |
| | | Type S [PtRh-Pt] | 2#0100 | 16#0C04 |
| | | Type J [Fe-CuNi IEC] | 2#0101 | 16#0C05 |
| | | Type L [Fe-CuNi DIN] | 2#0110 | 16#0C06 |
| | | Type T [Cu-CuNi IEC] | 2#0111 | 16#0C07 |
| | | Type K [NiCr-Ni] | 2#1000 | 16#0C08 |

| Measuring type | Coding (A) | Measuring range | Code (B) | MODE (256*A+B) |
|---|---|---|---|---|
| Thermocouple, linear, internal compensation / internal reference point | 2#1101 | Type B [PtRh-PtRh] | 2#0000 | 16#0D00 |
| | | Type N [NiCrSi-NiSi] | 2#0001 | 16#0D01 |
| | | Type E [NiCr-CuNi] | 2#0010 | 16#0D02 |
| | | Type R [PtRh-Pt] | 2#0011 | 16#0D03 |
| | | Type S [PtRh-Pt] | 2#0100 | 16#0D04 |
| | | Type J [Fe-CuNi IEC] | 2#0101 | 16#0D05 |
| | | Type L [Fe-CuNi DIN] | 2#0110 | 16#0D06 |
| | | Type T [Cu-CuNi IEC] | 2#0111 | 16#0D07 |
| | | Type K [NiCr-Ni] | 2#1000 | 16#0D08 |
| | | Type U [Cu-CuNi DIN] | 2#1001 | 16#0D09 |
| | | Type C | 2#1010 | 16#0D0A |
| | | Type TXK/XK(L) | 2#1011 | 16#0D0B |
| Thermocouple, linear, external compensation / reference point RTD(0) | 2#1110 | Type B [PtRh-PtRh] | 2#0000 | 16#0E00 |
| | | Type N [NiCrSi-NiSi] | 2#0001 | 16#0E01 |
| | | Type E [NiCr-CuNi] | 2#0010 | 16#0E02 |
| | | Type R [PtRh-Pt] | 2#0011 | 16#0E03 |
| | | Type S [PtRh-Pt] | 2#0100 | 16#0E04 |
| | | Type J [Fe-CuNi IEC] | 2#0101 | 16#0E05 |
| | | Type L [Fe-CuNi DIN] | 2#0110 | 16#0E06 |
| | | Type T [Cu-CuNi IEC] | 2#0111 | 16#0E07 |
| | | Type K [NiCr-Ni] | 2#1000 | 16#0E08 |
| | | Type U [Cu-CuNi DIN] | 2#1001 | 16#0E09 |
| | | Type C | 2#1010 | 16#0E0A |
| | | Type TXK/XK(L) | 2#1011 | 16#0E0B |
| Themocouple, dyn. ref. temp | 2#0111 | Type B [PtRh-PtRh] (variant 1) OMODE settings for SM modules (Page 396) | 2#0000 | 16#0700 |
| | | Type N [NiCrSi-NiSi] (variant 1) OMODE settings for SM modules (Page 396) | 2#0001 | 16#0701 |
| | | Type E [NiCr-CuNi] (variant 1) OMODE settings for SM modules (Page 396) | 2#0010 | 16#0702 |
| | | Type R [PtRh-Pt] (variant 1) OMODE settings for SM modules (Page 396) | 2#0011 | 16#0703 |

| Measuring type | Coding (A) | Measuring range | Code (B) | MODE (256*A+B) |
|---|---|---|---|---|
| | | Type S [PtRh-Pt] (variant 1) OMODE settings for SM modules (Page 396) | 2#0100 | 16#0704 |
| | | Type J [Fe-CuNi IEC] (variant 1) OMODE settings for SM modules (Page 396) | 2#0101 | 16#0705 |
| | | Type L [Fe-CuNi DIN] (variant 1) OMODE settings for SM modules (Page 396) | 2#0110 | 16#0706 |
| | | Type T [Cu-CuNi IEC] (variant 1) OMODE settings for SM modules (Page 396) | 2#0111 | 16#0707 |
| | | Type K [NiCr-Ni] (variant 1) OMODE settings for SM modules (Page 396) | 2#1000 | 16#0708 |

[1]: This measuring range is only supported by F channel blocks.

## Effect of the temperature coefficients on the measuring range

- Setting TC = 3850 at GOST Standard Pt10, Pt50, Pt100, Pt500 sets Bit 7 in the measuring range byte (0x40)
- Setting TC = 428 at GOST Standard Cu10, Cu50, Cu100 sets Bit 8 in the measuring range byte (0x80)

## Measuring range coding of the analog output modules

Depending on the measuring-range coding of the analog output modules, the parameter MODE_xx (measuring-range coding) corresponding to the channel must be specified in accordance with the table.

| Measuring type | Measuring range | MODE |
|---|---|---|
| Voltage | ± 5 V | 16#0106 |
| | 1 to 5 V | 16#0107 |
| | 0 to 10 V | 16#0108 |
| | ± 10 V | 16#0109 |
| Current | 0 mA to 20 mA | 16#0202 |
| | 4 to 20 mA | 16#0203 |
| | ± 20 mA | 16#0204 |
| HART interface | 4 mA to 20 mA | 16#070C |

### Measuring-Range Coding of the Digital Input and Output Modules

With digital input modules and digital output modules, there is no measuring type and no measuring range:
MODE = 16#FFFF (with DI)
MODE = 16#FFFE (with DO)

### Measuring range coding of the controller module

There is no measuring type and no measuring range for controller modules:

MODE = 16#FFFD

# 5.4 OMODE settings for SM modules

## OMODE structure

The table below shows the structure and meaning of the outputs OMODE_xx of data type DWORD:

| | | |
|---|---|---|
| Byte 3: | 16#80: Value status "valid value"<br>16#00: Value status "invalid value"<br>16#40: Value status "invalid value" | <br>(Channel error)<br>(Higher-level error) |
| Byte 2: | 16#01: Restart (OB 100) has been carried out<br>16#02: Measuring-range overshoot<br>16#04: Measuring range low limit exceeded<br>16#xy: Variant identifier at multiple MODE_ assignment (see below) | <br>(Channel-error diagnostics)<br>(Channel-error diagnostics) |
| Byte 1, 0 (low word): | MODE (see above) | |

Example:

16#80010203 = value status "valid value", restart has been carried out, current 4 mA to 20mA

## Mode: 16#090C and variant

| Variant | x | y | Measuring range |
|---|---|---|---|
| 0 | 0 | 0,1,2,4 | Ni 120 standard range |
| 1 | 1 | 0,1,2,4 | KTY84/110 |

## Mode: 16#090D and variant

| Variant | x | y | Measuring range |
|---|---|---|---|
| 0 | 0 | 0,1,2,4 | Ni 120 climate range |
| 1 | 1 | 0,1,2,4 | KTY84/130 |

## Mode: 16#07 (Coding A) and variant

| Variant | x | y | Measuring range |
|---|---|---|---|
| 0 | 0 | 0,1,2,4 | HART interface |
| 1 | 1 | 0,1,2,4 | Thermocouple, dynamic reference temperature |

## 5.5 MODE settings for PA devices

MODE_xx input parameters are available for a maximum of 32 slots of a PA field device. Their initial value is zero (no read/write access). You must set the combination selected from the options of the PROFIBUS PA 3.0 profile at the MODE_xx input of each slot channel xx:

| Block | I/O (parameters) (cyclic data)<br>permissible combination and sequence | Input (I)/Output (O)<br>(PLS view) | MODE 16#xxyy, O=xx I=yy |
|---|---|---|---|
| Analog input (PA_AI) | OUT | I | 16#0001 |
| Totalizer (PA_TOT) | TOTAL | I | 16#000F |
| Totalizer (PA_TOT) | TOTAL<br>SET_TOT | I<br>O | 16#070F |
| Totalizer (PA_TOT) | TOTAL<br>SET_TOT<br>MODE_TOT | I<br>O<br>O | 16#080F |
| Analog output (PA_AO) | SP | O | 16#0100 |
| Analog output (PA_AO) | SP<br>READBACK<br>POS_D | O<br>I<br>I | 16#0103 |
| Analog output (PA_AO) | SP<br>CHECK_BACK | O<br>I | 16#0104 |
| Analog output (PA_AO) | SP<br>READBACK<br>POS_D<br>CHECK_BACK | O<br>I<br>I<br>I | 16#0105 |
| Analog output (PA_AO) | RCAS_IN,<br>RCAS_OUT | O<br>I | 16#0206 |
| Analog output (PA_AO) | RCAS_IN,<br>RCAS_OUT,<br>CHECK_BACK | O<br>I<br>I | 16#0207 |
| Analog output (PA_AO) | SP<br>RCAS_IN<br>READBACK<br>RCAS_OUT<br>POS_D<br>CHECK_BACK | O<br>O<br>I<br>I<br>I<br>I | 16#0308 |
| Discrete input (PA_DI) | OUT_D | I | 16#0002 |
| Discrete output (PA_DO) | SP_D | O | 16#0400 |
| Discrete output (PA_DO) | SP_D<br>READBACK_D | O<br>I | 16#0409 |
| Discrete output (PA_DO) | SP_D<br>CHECKBACK_D | O<br>I | 16#040A |
| Discrete output (PA_DO) | SP_D<br>READBACK_D<br>CHECK_BACK_D | O<br>I<br>I | 16#040B |
| Discrete output (PA_DO) | RCAS_IN_D<br>RCAS_OUT_D | O<br>I | 16#050C |

| Block | I/O (parameters)<br>(cyclic data)<br>permissible combination and sequence | Input (I)/Output (O)<br><br>(PLS view) | MODE 16#xxyy,<br>O=xx I=yy |
|---|---|---|---|
| Discrete output (PA_DO) | RCAS_IN_D<br>RCAS_OUT_D<br>CHECK_BACK_D | O<br>I<br>I | 16#050D |
| Discrete output (PA_DO) | SP_D<br>RCAS_IN_D<br>READBACK_D<br>RCAS_OUT_D<br>CHECK_BACK_D | O<br>O<br>I<br>I<br>I | 16#060E |

## 5.6 Error Information of Output Parameter MSG_STAT

The messages can be disabled by setting input EN_MSG = FALSE (output MSG_STAT(_x) remains unchanged).

Block ALARM8_P(_x) is called in the acyclic OBs and in OB1 if message suppression is not enabled. Error information of ALARM_8P(_x) - messages cannot be output - is indicated at output parameter MSG_STAT(_x).

Details on the error information of output parameter MSG_STAT and on the acknowledgment word MSG_ACK(_x) of ALARM_8P are found in the Online Help of SF B35 (ALARM_8P).

## 5.7 Addressing

### Rules

If you do not use the CFC function "Generate module drivers", you must set the logical basic address of the module created with HW Config at the LADDR input parameter. If input SUBN_TYP = FALSE, the RACK of the module is connected to an integrated DP interface (distributed I/O device interface) of the CPU module by means of a line. Otherwise, you must set SUBN_TYP = TRUE.

The following points are generally to be observed for all SM, PA and FF blocks:

- The basic address of modules equipped only with inputs, i.e., modules which only write data to the input range of the CPU process image, can be fetched directly from HW Config;
  for example: The module SM 331 AI 8x12Bit 6ES7 331-7KF01-0AB0:

| Address input range (HW Config) | Address output range (HW Config) | LADDR (decimal/hex) |
|---|---|---|
| 512 | - | 512 / 16#0200 |

- The MSB (most significant bit) must be set in the basic address fetched from HW Config for modules equipped only with outputs, which means modules which only read data from the output range of the CPU process image;
  for example, module SM 332 AO 4x12Bit 6ES7 332-5HD01-0AB0:

| Address input range (HW Config) | Address output range (HW Config) | LADDR (decimal/hex) |
|---|---|---|
| - | 512 | -32256 / 16#8200 |

- The basic address of the input range must be set at input LADDR and the basic address of the output range from HW Config must be set at input LADDR1 for mixed modules, which means modules that write data to the input range and read data from the output range of the CPU process image;
  for example, module SM 323 DI/O 8x24V/05A 6ES7 323-1BH81-0AA0:

| Address input range (HW Config) | Address output range (HW Config) | LADDR (decimal/hex) | LADDR1 (decimal/hex) |
|---|---|---|---|
| 12 | 12 | 12 / 16# 000C | 12 / 16# 000C |

- For FF devices which only write data to the input range of the CPU process image, the lowest input address should be fetched from HW Config.
- For FF devices which only write data to the output range of the CPU process image, the lowest output address from HW Config should be set.
- For FF devices which write the data in the input range of the CPU process image and read from the output range of the CPU process image, you must set the lowest address of the input/output range from HW Config at the DADDR input.

| Address input range (HW Config) | Address output range (HW Config) | DADDR (decimal/hex) |
|---|---|---|
| 512 | - | 512 |
| | 512 | 16896 (16384 + 512)/ 16#4200 |
| 512 | 516 | 512 |
| 512 | 512 | 512 |
| 516 | 512 | 16896 (16384 + 512) / 16#4200 |

# 5.8 Message Classes

## Message classes

Message classes are used to group messages according to their cause. The following message classes are used in the SIMATIC process control system:

- Process messages triggered when process-specific monitoring values (for example, alarms, warnings, high/low tolerances, general process messages) are reached or exceeded.
- Process-control messages; output by the control system (system messages) or the I/O units (errors in the field), or for preventive maintenance.
- Requests for operator input which, in the case of certain operation sequences, draw the operator's attention to the necessity of an operator intervention (for example, request to acknowledge a stepping operation manually in order to enable transition) or operation logs.

### Table of message classes and their meaning

| Message class | Meaning | With acknowledgment |
|---|---|---|
| AH | Alarm high (High High Alarm) | Yes |
| AL | Alarm low (Low Low Alarm) | Yes |
| WH | Warning high (High Alarm) | Yes |
| WL | Warning low (Low Alarm) | Yes |
| TH | Tolerance high (Tolerance High) | Yes |
| TL | Tolerance low (Tolerance Low) | Yes |
| F | AS process control message (error) | Yes |
| S | AS process control message (fault) | Yes |
| S* | OS process control message (fault) | Yes |
| M | Preventive maintenance (Maintenance) | Yes |
| PM | Process message (Process Message) | Yes |
| PM | Operating message | No |
| OR | Operator request (Operator Request) | No |
| OM *1) | Operator message (Operation Message) | No |
| SA | Status AS | No |
| SO | Status OS | No |

*1) If the block is used for operation messages, the inputs I_1, ... have to be supplied with pulses. Assignment of the static value "1" would lead to multiple messages.

# 5.9 Dependencies

## Dependency on the FM_CO block

The FM_CO block (PCS 7 Basic Library) coordinates the reading of data records for the FM_CNT, FMCS_PID, FMT_PID, and READ355P blocks. Therefore, when installing these blocks you must ensure that the driver generator installs FM_CO in the fastest cyclic-interrupt OB of one of the blocks named above. This OB must not run slower than 30 s or faster than 25 ms.

## Notes on reading data records

In an ET 200M with n controller blocks, reading of data records is activated every n+1 cycles (ideally, when the installation sequence tallies with the interconnection).

This means that, in the worst case, a setpoint changed by means of a faceplate will only be viewed by the operator after n+1 cycles (e.g. this would amount to 17 s for OB 32 (1,000 ms) with 4 controller modules and 16 blocks).

The same applies for position feedback for the step controller. At the time of changeover (manual/auto), the manipulated variable LMN is set to an out-of-date value.

## Changing the visualization time

You can reduce the visualization time as follows:

- Install the controller blocks in a faster OB

  or

- Distribute the controller modules over several ET 200 stations.

# 5.10 Status displays

## 5.10.1 Maintenance Status of MS

### Layout of the maintenance status

The layout of the maintenance status MS (DWORD data type) is as follows:

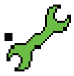
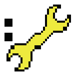
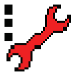
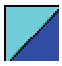
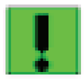
| | |
|---|---|
| Bit 0 to 7 | Display of the MS |
| Bit 8 to 15 | Display of the MS of the redundant partner |
| Bit 16 | 1 = Redundant partner available |
| Bit 17 | 0 = primary partner is master,<br>1 = redundant partner is master |
| Bit 18 | PDM-MS worse than device status |
| Bit 19 to 21 | Reserve |
| Bit 22 | PDM has detected status change |
| Bit 23 | Block takes part in the cyclical updating of PDM |
| Bit 24 to 27 | OS operation |
| Bit 28 to 31 | PDM-MS |

The MS is copied 1:1 to the output O_MS.

### Displayable Statuses

The maintenance status (MS) can display the following statuses, that are entered in Bit 0 to 7 or Bit 8 to 15 (for redundant partner):

| Bit number | | | | | | | | State | Symbol | Priority |
|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | 0 = highest priority |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Good |  | 9 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Passivated |  | 7 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Out of service |  | 6 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | At least one PV simulated |  | 5 |

| Bit number | | | | | | | | State | Symbol | Priority |
|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | 0 = highest priority |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Local operation/function test |  | 4 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Maintenance required |  | 3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | Maintenance request |  | 2 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | Maintenance alarm |  | 1 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Untested/unknown |  | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | Configuration changed |  | 8 |

**Note**

If the maintenance status is "untested/unknown", all other dynamic displays in the faceplates for Asset Management are not relevant to this instance.

## Maintenance Status of the Messages

The maintenance status is updated in the driver blocks by a message. The driver blocks generate a message with the following message classes:

| Message class | EventState bit | Symbol |
|---|---|---|
| AS process control message (S) = fault | 25 |   |
| AS process control message (F) = error | 24 |   |
| Preventive maintenance (M) = Maintenance | 23 |   |
| Status AS (SA) | 18 |  |

## Redundancy

In case of redundancy, several combinations of the displays are possible. See:

Status display for redundant components [Asset] (Page 406)

## 5.10.2 Status Display for Redundant Components [Asset]

### Status display icons

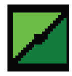
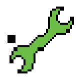
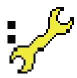
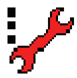
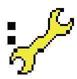
Redundant component A and redundant component B form the status display (maintenance state) for redundant components. In the following table, the icons of the status display are listed that result from this rule (the bit numbers not listed in the table are always = 0).

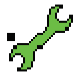
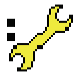
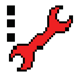
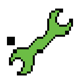
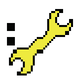
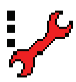
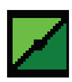
**Note**

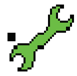
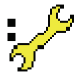
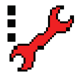
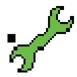
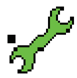
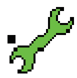
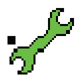
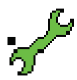
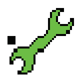
The status MS = 9, configuration changed, is for the redundant components and is therefore not listed here.

PV = process value

| Bit number | | | | | | | | State | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 11 | 10 | 9 | 8 | 3 | 2 | 1 | 0 | Redundant component A | Redundant component B | Status display icon |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Good | Good | Good |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | Good | Passivated | Good |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | Good | Out of service | Maintenance request |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | Good | At least 1 PV simulated | Good |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | Good | Local operation/ function test | Good |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | Good | Maintenance required | Maintenance required |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | Good | Maintenance request | Maintenance request |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Good | Maintenance alarm | Maintenance request |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Good | Untested/ unknown | Good |
| | | | | | | | | | | |

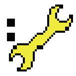
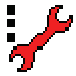
| Bit number | | | | | | | | State | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Passivated | Good | Good |  |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | Passivated | Passivated | Passivated |  |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | Passivated | Out of service | Out of service |  |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | Passivated | At least 1 PV simulated | At least 1 PV simulated |  |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | Passivated | Local operation/ function test | Local operation/ function test |  |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | Passivated | Maintenance required | Maintenance required |  |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | Passivated | Maintenance request | Maintenance request |  |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | Passivated | Maintenance alarm | Maintenance alarm |  |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Passivated | Untested/ unknown | Passivated |  |
| | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Out of service | Good | Maintenance request |  |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | Out of service | Passivated | Out of service |  |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | Out of service | Out of service | Out of service |  |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | Out of service | At least 1 PV simulated | At least 1 PV simulated |  |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | Out of service | Local operation/ function test | Local operation/ function test |  |

| Bit number | | | | | | | | State | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | Out of service | Maintenance required | Maintenance required |  |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | Out of service | Maintenance request | Maintenance request |  |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | Out of service | Maintenance alarm | Maintenance alarm |  |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Out of service | Untested/ unknown | Out of service |  |
| | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | At least 1 PV simulated | Good | Good |  |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | At least 1 PV simulated | Passivated | At least 1 PV simulated |  |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | At least 1 PV simulated | Out of service | At least 1 PV simulated |  |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | At least 1 PV simulated | At least 1 PV simulated | At least 1 PV simulated |  |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | At least 1 PV simulated | Local operation/ function test | Local operation/ function test |  |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | At least 1 PV simulated | Maintenance required | Maintenance required |  |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | At least 1 PV simulated | Maintenance request | Maintenance request |  |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | At least 1 PV simulated | Maintenance alarm | Maintenance alarm |  |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | At least 1 PV simulated | Untested/ unknown | At least 1 PV simulated |  |
| | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Local operation/ function test | Good | Good |  |

| Bit number | | | | | | | | State | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | Local operation/ function test | Passivated | Local operation/ function test |  |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | Local operation/ function test | Out of service | Local operation/ function test |  |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | Local operation/ function test | At least 1 PV simulated | Local operation/ function test |  |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | Local operation/ function test | Local operation/ function test | Local operation/ function test |  |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | Local operation/ function test | Maintenance required | Maintenance required |  |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | Local operation/ function test | Maintenance request | Maintenance request |  |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | Local operation/ function test | Maintenance alarm | Maintenance alarm |  |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Local operation/ function test | Untested/ unknown | Local operation/ function test |  |
| | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Maintenance required | Good | Maintenance required |  |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | Maintenance required | Passivated | Maintenance required |  |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | Maintenance required | Out of service | Maintenance required |  |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | Maintenance required | At least 1 PV simulated | Maintenance required |  |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | Maintenance required | Local operation/ function test | Maintenance required |  |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | Maintenance required | Maintenance required | Maintenance required |  |

| Bit number | | | | | | | | State | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | Maintenance required | Maintenance request | Maintenance request | |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | Maintenance required | Maintenance alarm | Maintenance alarm | |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | Maintenance required | Untested/ unknown | Maintenance required | |
| | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | Maintenance request | Good | Maintenance request | |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | Maintenance request | Passivated | Maintenance request | |
| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | Maintenance request | Out of service | Maintenance request | |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | Maintenance request | At least 1 PV simulated | Maintenance request | |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | Maintenance request | Local operation/ function test | Maintenance request | |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | Maintenance request | Maintenance required | Maintenance request | |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | Maintenance request | Maintenance request | Maintenance request | |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | Maintenance request | Maintenance request | Maintenance alarm | |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | Maintenance request | Untested/ unknown | Maintenance request | |
| | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | Maintenance alarm | Good | Maintenance request | |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | Maintenance alarm | Passivated | Maintenance alarm | |

| Bit number | | | | | | | | State | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | Maintenance alarm | Out of service | Maintenance alarm | |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | Maintenance alarm | At least 1 PV simulated | Maintenance alarm | |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | Maintenance alarm | Local operation/ function test | Maintenance alarm | |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | Maintenance alarm | Maintenance required | Maintenance alarm | |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | Maintenance alarm | Maintenance request | Maintenance alarm | |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | Maintenance alarm | Maintenance alarm | Maintenance alarm | |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | Maintenance alarm | Untested/ unknown | Maintenance alarm | |
| | | | | | | | | | | | |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Untested/ unknown | Untested/ unknown | Untested/ unknown | |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Untested/ unknown | Good | Good | |
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | Untested/ unknown | Passivated | Passivated | |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | Untested/ unknown | Out of service | Out of service | |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | Untested/ unknown | At least 1 PV simulated | At least 1 PV simulated | |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | Untested/ unknown | Local operation/ function test | Local operation/ function test | |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | Untested/ unknown | Maintenance required | Maintenance required | |

| Bit number | | | | | | | | State | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | Untested/ unknown | Maintenance request | Maintenance request |  |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | Untested/ unknown | Maintenance alarm | Maintenance alarm |  |

## 5.10.3 PA field device status and diagnostics information

### PA-Field-Device Status

| PA Status<br>Limits (Bit 1 + 0 irrelevant) | | Meaning of PA Status | Message | Coding MS |
|---|---|---|---|---|
| Quality<br>(bit 7 + 6) | Substatus<br>(bits 5 – 2) | | M = Message<br>Q = Must be acknowledged | |
| 10 | 0000 | Good | | 0 |
| 10 | 0001 | Good, configuration has been changed | M | 0 |
| 10 | 0010 | Good, active warning | | 0 |
| 10 | 0011 | Good, active interrupt | | 0 |
| 10 | 0100 | Good, unacknowledged configuration change | | 0 |
| 10 | 0101 | Good, unacknowledged warning | | 0 |
| 10 | 0110 | Good, unacknowledged interrupt | | 0 |
| 10 | 1000 | Good, device switches to fail-safe position | M | 0 |
| 10 | 1001 | Good, maintenance required<br>(more diagnostic data available) | Q | 5 |
| 10 | 1010 | Good, maintenance request pending<br>(more diagnostic data available) | Q | 6 |
| 10 | 1111 | Good, self-test completed | | 0 |
| 01 | 0000 | Uncertain, not specified | Q | 6 |
| 01 | 0001 | Uncertain, last valid value | Q | 7 |
| 01 | 0010 | Uncertain, substitute value | Q | 7 |
| 01 | 0011 | Uncertain, initial value | | 0 |
| 01 | 0100 | Uncertain, measured value generation incorrect | Q | 6 |
| 01 | 0101 | Uncertain, value outside defined range | Q | 6 |
| 01 | 0110 | Uncertain, signal source uncertain | Q | 6 |
| 01 | 0111 | Uncertain, configuration error | Q | 6 |
| 01 | 1000 | Uncertain, measured value simulated | Q | 3 |
| 01 | 1001 | Uncertain, sensor calibration | Q | 6 |
| 01 | 1010 | Uncertain, maintenance request<br>(more diagnostic data available) | Q | 6 |
| 01 | 1100 | Uncertain, simulation started | M | 3 |
| 01 | 1101 | Uncertain, simulation ended | M | 0 |
| 01 | 1110 | Uncertain, process-related, no maintenance | M | 0 |
| 00 | 0000 | Bad, not specified | Q | 7 |
| 00 | 0001 | Bad, configuration error | Q | 7 |
| 00 | 0010 | Bad, not connected | Q | 7 |
| 00 | 0011 | Bad, device error | Q | 7 |
| 00 | 0100 | Bad, sensor error | Q | 7 |
| 00 | 0101 | Bad, no connection - last valid value | Q | 7 |

| PA Status<br>Limits (Bit 1 + 0 irrelevant) | | Meaning of PA Status | Message | Coding MS |
|---|---|---|---|---|
| 00 | 0110 | Bad, no connection - no valid value | Q | 7 |
| 00 | 0111 | Bad, device out of service | Q | 2 |
| 00 | 1000 | Bad, device out of service<br>(no diagnostic data, passivated) | M | 1 |
| 00 | 1001 | Bad, maintenance alarm<br>(more diagnostic data available) | Q | 7 |
| 00 | 1010 | Bad, process-related, no maintenance | M | 0 |
| 00 | 1111 | Bad, local operation/functional check | M | 4 |

You will find additional information about status icons for the maintenance status (MS) in the "Maintenance status MS (Page 403)" section.

## Diagnostic information

The structure of the PA_DIAG parameter is as follows:

| Byte | Bit position | PROFIBUS - diagnostic significance | Message | MS |
|---|---|---|---|---|
| 0 | 0 | Electronic hardware failure | Q | 7 |
| | 1 | Mechanical hardware failure | Q | 7 |
| | 2 | Excess motor temperature | Q | 6 |
| | 3 | Excess temperature at electronic circuit | Q | 6 |
| | 4 | Memory error | Q | 7 |
| | 5 | Measurement failure | Q | 7 |
| | 6 | Device not initialized (no auto-calibration) | Q | 0 |
| | 7 | Auto-calibration error | | 7 |
| 1 | 0 | Zero error (limit position) | Q | 6 |
| | 1 | No power supply (electr. pneum.) | Q | 7 |
| | 2 | Invalid configuration | Q | 7 |
| | 3 | Warm start executed | M | 0 |
| | 4 | Complete restart executed | M | 0 |
| | 5 | Maintenance required | Q | 5 |
| | 6 | Invalid identifier | Q | 7 |
| | 7 | Invalid ID number | Q | 7 |

| Byte | Bit position | PROFIBUS - diagnostic significance | Message | MS |
|---|---|---|---|---|
| 2 | 0 | Device error | Q | 7 |
| | 1 | Maintenance requested | Q | 5 |
| | 2 | Device is in function test or in simulation or is under local operator control (maintenance) | | 0 |
| | 3 | The process conditions do not allow the valid values to be returned; this is set when the quality is “uncertain, process-related, no maintenance” or “bad, process-related, no maintenance”. | | 0 |
| | 4 - 7 | Reserved for PNO, default 0 | | |
| 3 | 0-4 | Reserved for PNO use | | |
| | 5 | = 0: for devices of the corresponding profile | | |
| | 6 | = 0: for devices of the corresponding profile | | |
| | 7 | = 0: no further information available | | |
| | | = 1: Further diagnostic information available in DIAGNOSIS_EXTENSION | | |

# 5.11 Text libraries

## 5.11.1 Text Library for MOD_PAL0, MOD_PAX0

The following table lists the text-library message texts and their numbers for the blocks MOD_PAL0 (FB99) and MOD_PAX0 (FB112):

| Text no. | Message text |
|---|---|
| 1 | Warm restart |
| 2 | Cold restart |
| 3 | PA field device diagnostics |
| 4 | Memory error |

## 5.11.2 Text Library for PADP_L00, PADP_L01, PADP_L02

The following table lists the text-library message texts and their numbers for the blocks PADP_L00 (FB 109), PADP_L01 (FB 110), PADP_L02 (FB 111):

| Text No. | Message Text |
|---|---|
| 1 | Module error |
| 2 | Wrong module |
| 3 | Module missing |

## 5.11.3 Text Library for DREP, DREP_L

The following table lists the text-library message texts and their numbers for the blocks DREP (FB 113) and DREP_L (FB 125):

| Text No. | Message Text |
|---|---|
| 1 | Failure |
| 2 | Diagnostics |

### 5.11.4 Text library for MOD_1, MOD_2, MOD_3, MOD_64, MOD_D2, MOD_CP

The following table lists the message texts and their text numbers from the text library for the MOD_1 (FB 91) / MOD_2 (FB 92) / MOD_3 (FB 95) / MOD_64 (FB 137) / MOD_D2 (FB 94) / MOD_CP (FB 98) blocks :

| Text No. | Message Text | Remark |
|---|---|---|
| 1 | Parameter assignment error | |
| 2 | Common mode error | |
| 3 | Short-circuit to P | |
| 4 | Short circuit to M | |
| 5 | Wire break | |
| 6 | Reference channel error | |
| 7 | Measuring range violation low | |
| 8 | Measuring range violation high | |
| 9 | Load voltage missing | |
| 10 | Chassis ground error | |
| 11 | Sensor supply missing | |
| 12 | Excess temperature | |
| 13 | Module OK | |
| 14 | Internal error | |
| 15 | External error | |
| 16 | External auxiliary voltage missing | |
| 17 | Front connector missing | |
| 18 | No configuration | |
| 19 | Wrong module parameters | |
| 20 | Wrong/missing user module | |
| 21 | Communication error | |
| 22 | Operating mode RUN/STOP | RUN: going STOP: coming |
| 23 | Timeout | |
| 24 | Failed module int. supply voltage | |
| 25 | Battery depleted | |
| 26 | Total backup failure | |
| 27 | CPU failure | |
| 28 | EPROM error | |
| 29 | RAM error | |
| 30 | ADC/DAC error | |
| 31 | Fuse tripped | |
| 32 | Process interrupt lost | |
| 33 | Removed | |
| 34 | Plugged | |
| 35 | Wrong module type plugged | |
| 36 | Faulty module inserted | |

| Text No. | Message Text | Remark |
|---|---|---|
| 37 | Module inserted (parameter error) | |
| 38 | Chatter error | |
| 39 | Changeover contact diagnostics | |
| 40 | CIR parameter assignment | |
| 41 | CIR parameter assignment not successful | |

### 5.11.5 Text Library for MOD_D1

The following table lists the text-library message texts and their numbers for the block MOD_D1 (FB 93):

| Text no. | Message text | Remark |
|---|---|---|
| 1 | Parameter assignment error | |
| 2 | Common mode error | |
| 3 | Short-circuit to P | |
| 4 | Short circuit to M | |
| 5 | Wire break | |
| 6 | Reference channel error | |
| 7 | Measuring range violation low | |
| 8 | Measuring range violation high | |
| 9 | Load voltage missing | |
| 10 | Chassis ground error | |
| 11 | Sensor supply missing | |
| 12 | Excess temperature | |
| 13 | Module OK | |
| 14 | Internal error | |
| 15 | External error | |
| 16 | External auxiliary voltage missing | |
| 17 | Front connector missing | |
| 18 | No configuration | |
| 19 | Wrong parameter in module | |
| 20 | Wrong/missing user module | |
| 21 | Communication error | |
| 22 | Operating mode RUN/STOP | RUN: going STOP: coming |
| 23 | Timeout | |
| 24 | Failed module int. supply voltage | |
| 25 | Battery depleted | |
| 26 | Total backup failure | |
| 27 | CPU failure | |
| 28 | EPROM error | |
| 29 | RAM error | |
| 30 | ADC/DAC error | |
| 31 | Fuse tripped | |
| 32 | Process interrupt lost | |
| 33 | Removed | |
| 34 | Plugged | |
| 35 | Wrong module type plugged | |
| 36 | Faulty module inserted | |
| 37 | Module inserted (parameter assignment error) | |

| Text no. | Message text | Remark |
|---|---|---|
| 38 | Chatter error | |
| 39 | Changeover contact diagnostics | |
| 40 | Sensor or load voltage loss | |
| 41 | Faulty fuse | |
| 42 | AI error hardware | |
| 43 | AI wire break | |
| 44 | AI below measuring range | |
| 45 | AI above measuring range | |
| 46 | AO wire break | |
| 47 | AO short-circuit | |
| 48 | CIR parameter assignment | |
| 49 | CIR parameter assignment not successful | |
| 50 | Signal A faulty | |
| 51 | Signal B faulty | |
| 52 | Signal N faulty | |
| 53 | Faulty value supplied to the channels | |
| 54 | Sensor supply 5.2 V/8.2 V faulty | |
| 55 | Sensor supply 24 V faulty | |
| 56 | Namur sensor signal line error | |
| 57 | Readback error | |
| 58 | Power supply 1: Error | |
| 59 | Power supply 2: Error | |
| 60 | Actuator OFF | |
| 61 | Discrepancy error | |

### 5.11.6 Text library for MOD_D3

The following table lists the text library message texts and their numbers for the block MOD_D3 (FB134):

| Text no. | Message text | Remark |
|---|---|---|
| 1 | Parameter assignment error | |
| 2 | Common mode error | |
| 3 | Short-circuit to P | |
| 4 | Short circuit to M | |
| 5 | Wire break | |
| 6 | Reference channel error | |
| 7 | Measuring range violation low | |
| 8 | Measuring range violation high | |
| 9 | Load voltage missing | |
| 10 | Chassis ground error | |
| 11 | Sensor supply missing | |
| 12 | Excess temperature | |
| 13 | Module OK | |
| 14 | Internal error | |
| 15 | External error | |
| 16 | External auxiliary voltage missing | |
| 17 | Front connector missing | |
| 18 | No configuration | |
| 19 | Wrong parameter in module | |
| 20 | Wrong/missing user module | |
| 21 | Communication error | |
| 22 | Operating mode RUN/STOP | RUN: going STOP: coming |
| 23 | Timeout | |
| 24 | Failed module int. supply voltage | |
| 25 | Battery depleted | |
| 26 | Total backup failure | |
| 27 | CPU failure | |
| 28 | EPROM error | |
| 29 | RAM error | |
| 30 | ADC/DAC error | |
| 31 | Fuse tripped | |
| 32 | Process interrupt lost | |
| 33 | Removed | |
| 34 | Plugged | |
| 35 | Wrong module type plugged | |
| 36 | Faulty module inserted | |
| 37 | Module inserted (parameter assignment error) | |

| Text no. | Message text | Remark |
|---|---|---|
| 38 | Chatter error | |
| 39 | Changeover contact diagnostics | |
| 40 | Sensor or load voltage loss | |
| 41 | Faulty fuse | |
| 42 | AI error hardware | |
| 43 | AI wire break | |
| 44 | AI below measuring range | |
| 45 | AI above measuring range | |
| 46 | AO wire break | |
| 47 | AO short-circuit | |
| 48 | CIR parameter assignment | |
| 49 | CIR parameter assignment not successful | |
| 50 | Signal A faulty | |
| 51 | Signal B faulty | |
| 52 | Signal N faulty | |
| 53 | Faulty value supplied to the channels | |
| 54 | Sensor supply 5.2 V/8.2 V faulty | |
| 55 | Sensor supply 24 V faulty | |
| 56 | Namur sensor signal line error | |
| 57 | Readback error | |
| 58 | Power supply 1: Error | |
| 59 | Power supply 2: Error | |
| 60 | Actuator OFF | |
| 61 | Undervoltage | |
| 62 | Overvoltage | |
| 63 | Overload | |
| 64 | Reserve | |
| 65 | Hardware interrupt | |
| 66 | Actuator warning | |
| 67 | Safety shutdown | |
| 68 | Ambiguous error | |
| 69 | Error 1 in actuator/sensor | |
| 70 | Error 2 in actuator/sensor | |
| 71 | Channel temporarily not available | |
| 72 | Channel is being calibrated | |
| 73 | Discrepancy error | |
| 74 | Overcurrent at output driver | |
| 75 | P-output driver defective | |
| 76 | Internal P short circuit | |
| 77 | External P short circuit/cross fault | |
| 78 | Internal error in the read circuit/test circuit | |

| Text no. | Message text | Remark |
|---|---|---|
| 79 | Short circuit of the output to M or output driver defective | |
| 80 | Switching frequency exceeded | |
| 81 | PROFIsafe communications error | |

## 5.11.7 Text Library for MOD_MS

The following table lists the text-library message texts and their numbers for the block MOD_MS (FB 96):

| Text No. | Message Text | Remark |
|---|---|---|
| 1 | Parameter assignment error | |
| 2 | Common mode error | |
| 3 | Short-circuit to P | |
| 4 | Short circuit to M | |
| 5 | Wire break | |
| 6 | Reference channel error | |
| 7 | Measuring range violation low | |
| 8 | Measuring range violation high | |
| 9 | Load voltage missing | |
| 10 | Chassis ground error | |
| 11 | Sensor supply missing | |
| 12 | Excess temperature | |
| 13 | Module OK | |
| 14 | Internal error | |
| 15 | External error | |
| 16 | External auxiliary voltage missing | |
| 17 | Front connector missing | |
| 18 | No configuration | |
| 19 | Wrong module parameters | |
| 20 | Wrong/missing user module | |
| 21 | Communication error | |
| 22 | Operating mode RUN/STOP | RUN: going STOP: coming |
| 23 | Timeout | |
| 24 | Failed module int. supply voltage | |
| 25 | Battery depleted | |
| 26 | Total backup failure | |
| 27 | CPU failure | |
| 28 | EPROM error | |
| 29 | RAM error | |
| 30 | ADC/DAC error | |
| 31 | Fuse tripped | |
| 32 | Process interrupt lost | |
| 33 | Removed | |
| 34 | Plugged | |
| 35 | Wrong module type plugged | |
| 36 | Faulty module inserted | |
| 37 | Module inserted (parameter error) | |

| Text No. | Message Text | Remark |
|---|---|---|
| 38 | Chatter error | |
| 39 | Changeover contact diagnostics | |
| 40 | Sensor or load voltage loss | |
| 41 | Faulty fuse | |
| 42 | Undervoltage | |
| 43 | Overvoltage | |
| 44 | Actuator warning | |
| 45 | Actuator OFF | |
| 46 | Safety-related shutdown | |
| 47 | Unknown error | |
| 48 | Short-circuit | |
| 49 | Error | |
| 50 | CIR parameter assignment | |
| 51 | CIR parameter assignment not successful | |

### 5.11.8 Text Library for OB_BEGIN

The following table lists the text-library message texts and their numbers for the block OB_BEGIN (FB 100):

| Text No. | Message Text | Remark |
|---|---|---|
| 1 | Nested stack error | Error code B#16#71: |
| 2 | Master control relay stack error | Error code B#16#72: |
| 3 | Max. nesting depth for synchronous errors exceeded | Error code B#16#73 |
| 4 | U-stack nesting depth exceeded | Error code B#16#74 |
| 5 | B-stack nesting depth exceeded | Error code B#16#75 |
| 6 | Local data allocation error | Error code B#16#76 |
| 7 | Unknown opcode | Error code B#16#78 |
| 8 | Code length error | Error code B#16#7A |

# Index

## E



## F



## G



## I



## M

## O



## P