Section **Page**

Section **Page**

1.1 Introduction

Diagnostic DFBs can be used with PL7 PRO or PL7 Junior and comprise :

- Application diagnostic DFBs which are used to monitor the process via the application program :
  - monitor a PL7 equation
  - monitor the reaction time of the process to a command
  - monitor the safety conditions
  - monitor the I/O and the ASI bus
- Application control and diagnostic DFBs which are used to control and monitor application components (EPO) :
  - check sensor data
  - check control requests to an actuator
  - monitor the duration of a movement
  - store the minimum and maximum durations of a movement
  - learn the duration of a movement
  - control an actuator

Compatibility : TSX57/PCX57/PMX57 software version > V3.3.

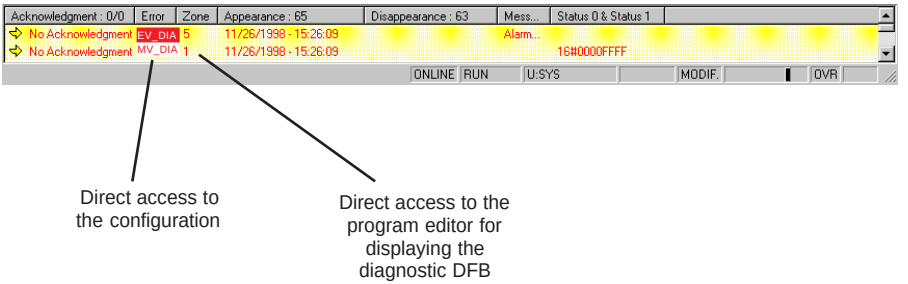
The library is made up of the following DFBs :

EV_DIA	Monitors the state of 2 bits without taking a time factor into account
MV_DIA	Monitors the state of 2 bits without taking a time factor into account with the possibility of monitoring the progress of a movement (change of state of a bit within a defined time limit)
NEPO_DIA **TEPO_DIA**	Monitors, controls and diagnoses an application component
IO_DIA	Diagnoses all I/O modules
ASI_DIA	Diagnoses an ASI I/O module
ALRM_DIA	Interface with the diagnostic buffer (stores errors)

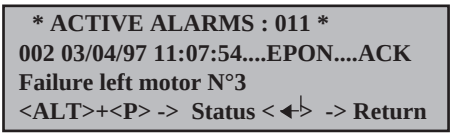
Error message
Each DFB has its own error message which may be standard or customizable depending on the DFB type.

Displaying error messages

1. The Viewer integrated in the PL7 Pro and PL7 ProDyn workshops is used to display diagnostic messages clearly (see Runtime Screens documentation).



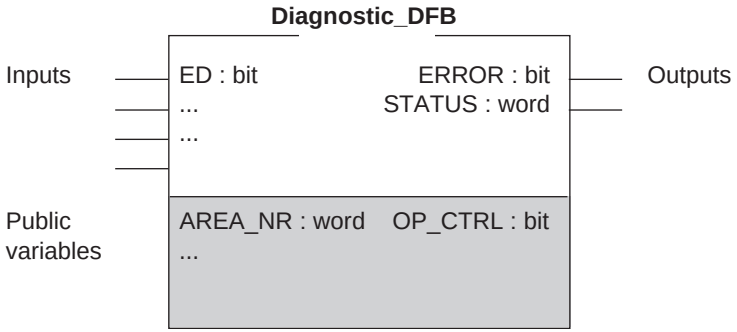
2. A diagnostics Viewer is also available with CCX17 V2.5 (see CCX 17 Operator Panel documentation).



1.2 General Characteristics of a Diagnostic DFB

1.2-1 Presentation

Diagnostic DFBs have the following structure :



Depending on the type of DFB, some parameters may be missing.
These DFBs cannot be modified by the user.
The descriptive file gives details on the DFB.

1.2-2 Description of the Parameters

Input parameters

Parameter	Type	Access	Description
ED	bit	R (1)	Monitoring enable bit : If ED = 0, the DFB inputs are not monitored. The default value of ED is 0.

(1) : Access by program

Output parameters

Parameter	Type	Access	Description
ERROR	bit	R (1)	Error bit. This bit is set to 1 as soon as an error occurs. This bit is set to 0 if the ED input returns to 0 or if no errors occur.
STATUS	word	R (1)	This word indicates the type of error. This word is at 0 if there are no errors. This word is set to 0 if the ED input returns to 0 or if no errors occur.

(1) : Access by program

Public variables

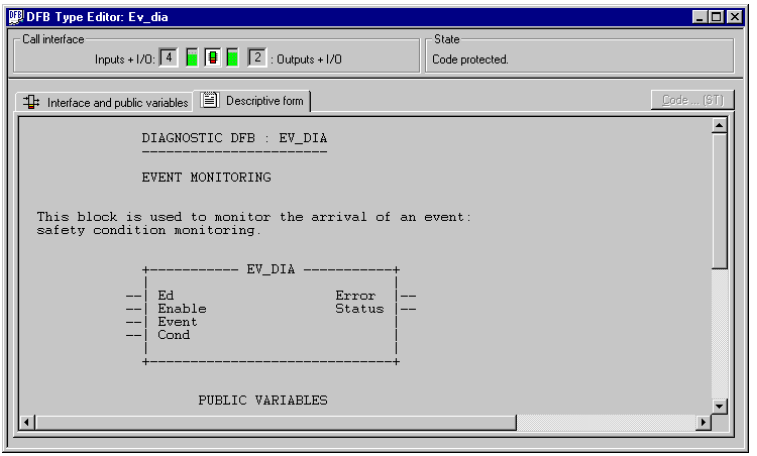
Parameter	Type	Access	Description
AREA_NR	word	R (1)	This word is used to specify which area of the control system is monitored by the diagnostic DFB. Example : Machining : no.1 Milling : no. 2 Tapping : no. 3 AREA_ NR must have the value 1, 2 or 3 so that the user can identify the faulty part of the control system. The above areas should be divided in the same way as the function modules. AREA_ NR can take a value from 0 to 15 (the default value is 0).
OP_CTRL	bit	R (1)	This bit indicates whether or not acknowledgment of the DFB instance by the operator is required. OP_CTRL = 0 : no acknowledgment by the operator OP_CTRL = 1 : acknowledgment by the operator The default value of OP_CTRL is 0.

(1) : Access by program

Note : public variables are only taken into account on a cold restart.

1.2-3 Descriptive File

Each diagnostic DFB has a descriptive file describing the function of the DFB and its parameters (inputs, outputs and public variables).
This file is accessed by double-clicking on a DFB type in the Application Browser and then double-clicking on the "Descriptive File" tab in the DFB editor.



1.3 Programming a Diagnostic DFB Using PL7

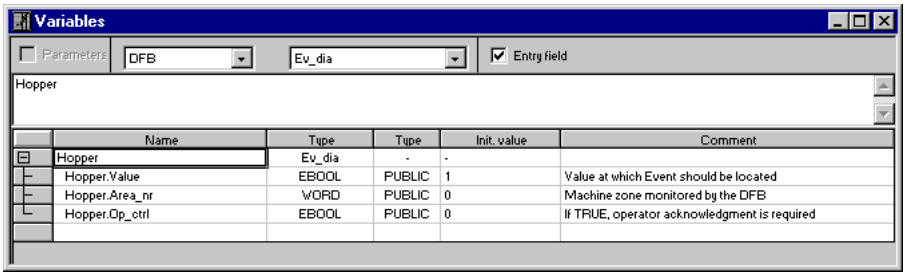
1.3-1 Configuring the Diagnostics Option

1. Select the Station directory in the Application Browser.
2. Access the **Station Properties** dialog box (click with the right mouse button on the Station item in the Application Browser and select from the Properties menu).
3. Select the **Diagnostics** tab.
4. Check the **Activate diagnostics in the application** box.

Activating the diagnostics option reserves a diagnostic buffer for storage of alarms (or errors) generated from diagnostic DFBs.

1.3-2 Declaring DFBs

Before using a DFB in an application program, the user must :
1. Import the DFB binary file (.UFB file), using the **Import Binary File** shortcut menu, from the **DIAG subdirectory** which is located in the PL7 installation directory (for example, C:\PL7\PL7PRO33\DIAG).
2. Declare an instance of the DFB in the PL7 variables editor.
For more information on DFBs, see the PL7 Micro/Junior/Pro Reference Manual, part A, section 6.
Example : “Hopper” is an instance of the EV_DIA DFB.



1.3-3 Customizing Error Messages

The user can customize the error message displayed with each error of a diagnostic DFB instance (except for the IO_DIA and ASI_DIA DFBs). This is done by changing the comment associated with the instance declared in the variables editor.
Example : “Hopper” is an instance of the EV_DIA DFB.
The comment for “Hopper” is “Silo empty or weighing hopper open”. This will be the user error message for the “Hopper” instance.

Note : The error messages are standard for the IO_DIA and ASI_DIA DFBs.

1.3-4 Rules for Programming DFBs

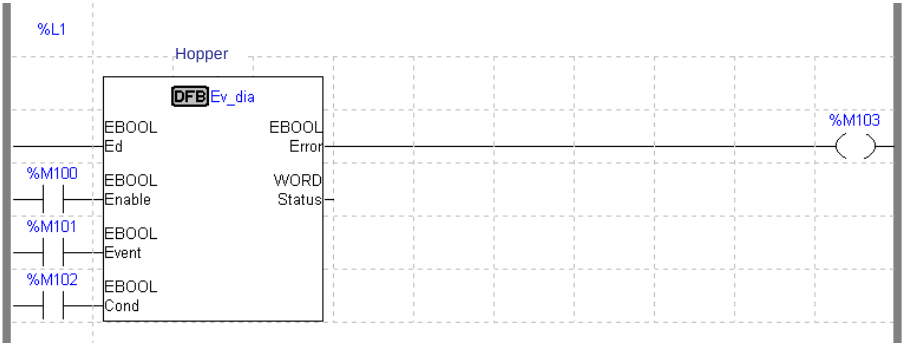
All diagnostic DFBs can be programed in any program module (Main, SR or section) in Ladder (LD), Structured Text (ST) and Instruction List (IL) language.

Rules

1. A diagnostic DFB must be executed in the MAST task in order to manage the operating modes.
2. It is strongly recommended that a diagnostic DFB instance is only programmed once in the application.
3. To execute a diagnostic DFB :
   - The DFB must be called (the program element to which it is assigned must be executed).
   - The ED input must be at 1.
4. The label is compulsory on the rung, statement or sequence containing the diagnostic DFB.

LD language

Programming example



ST language

The programming syntax is as follows :

%Li :	**label**
Inst (I1,..., In, O1,...,On);	
%Li :	**label**
Inst	Name of a diag DFB instance
I1,..., In	Inputs of the diag DFB
O1,...,On	Variables associated with the outputs of the diag DFB

Example : programming the previous example in ST language

Inputs :
ED : always true -> TRUE
ENABLE : Filling
EVENT : Closed
COND : Level

Outputs :
Error : Klaxon

! %L1 Hopper (TRUE, Filling, Closed, Level, Klaxon,);

IL language
The programming syntax is as follows :

[Inst (I1,..., In, O1,...,On)]

Inst	Name of a diag DFB instance
I1,..., In	Inputs of the diag DFB
O1,...,On	Variables associated with the outputs of the diag DFB

Example : programming the previous example in IL language

Inputs :	Outputs :
ED : always true -> TRUE	Error : Klaxon
ENABLE : Filling	
EVENT: Closed	
COND : Level	

! **%L1 LD TRUE**
[hopper (TRUE, Filling, Closed, Level, Klaxon,)]

1.4 Error Messages of Diagnostic DFBs

Each DFB has its own error message which may be standard or customizable depending on the DFB type.
The size of the error messages is limited to 40 characters.

Standard error messages

EV_DIA	"EVENT<>VALUE and/or COND < > 1"
MV_DIA	"EVENT<>VALUE,COND,EVENT_T0,EVENT_T1 < > 1"
NEPO_DIA **TEPO_DIA**	"Configuration or operating part error"
IO_DIA	"I/O error"
ASI_DIA	Standard error message depending on the type of error : "module or bus error" "At least 1 slave absent" "At least 1 slave not configured" "At least 1 slave failed"
ALRM_DIA	"COND1 < > 1 or COND0 < > 0"

User error messages
A user error message can be defined by including a comment with the DFB instance in the PL7 variables editor (except for the IO_DIA and ASI_DIA DFBs).

Caution! Only the first 40 characters are taken into account when creating the **user error message**.

Rules

1. Only the first 40 characters are taken into account when creating the user error message.
2. There are no user error messages for the IO_DIA and ASI_DIA DFBs, only standard error messages.
3. The Viewer displays the user error message if there is one, otherwise it displays the standard error message.
4. The standard error message is identical for all DFB instances.
5. The user error message can be different for each DFB instance.

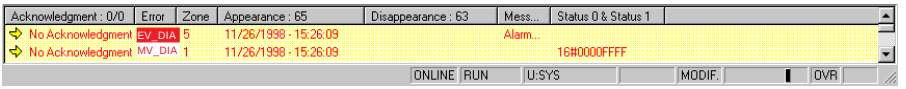
System information

System bits and words provide information on diagnostics :

%S101 = 1 diagnostic buffer configured
%S102 = 1 diagnostic buffer full
%SW162 number of errors in the diagnostic buffer

Caution : If the diagnostic buffer cannot register an error, this error is lost and bit %S102 changes to 1.

1.5 Displaying Error Messages in the Integrated Viewer



Each line displayed in the viewer corresponds to a message which can contain the following information :

- An icon and text indicating the message status : the message needs to be acknowledged, has been acknowledged or does not require acknowledgment
- The faulty DFB type (EV_DIA, MV_DIA, NEPO_DIA, ALRM_DIA, etc) and the location of the DFB instance in the program
- The area in which the error originated (AREA_NR public variable)
- The date and time at which the error appeared
- The date and time at which the error disappeared
- The message associated with the error
- The status value at the time of the error

Appearance of a message

Messages appear systematically at the end of the list.

Display management

Two elements can be configured :

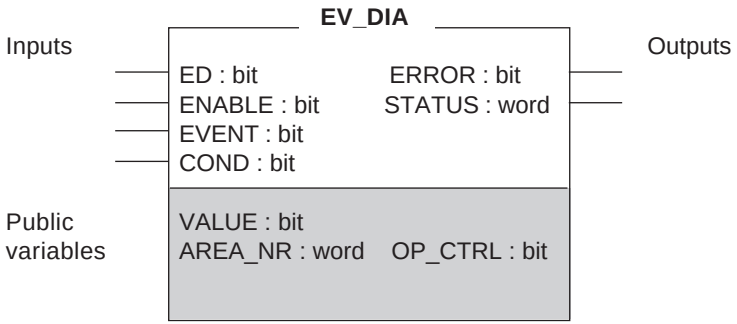
- The color of messages (text and background)
- The flashing associated with a message requiring acknowledgment

For more information, see Runtime Screens documentation.

2.1 General

The EV_DIA DFB is used to monitor the state of 2 bits without taking a time factor into account.

2.2 Presentation of EV_DIA



2.3 Description of the Parameters

Input parameters

Parameter	Type	Access	Description
ED	bit	R (1)	DFB activation bit If ED = 0, the EVENT and COND inputs are not monitored. The default value of ED is 0.
ENABLE	bit	R (1)	Monitoring enable bit If ENABLE = 0, only the COND input is monitored. If ENABLE = 1, the COND and EVENT inputs are monitored. The default value of ENABLE is 0.
EVENT	bit	R (1)	Input bit to be monitored If the DFB is executed and if ENABLE = 1, the DFB monitors that : • The EVENT input takes the value specified in the VALUE public variable • The EVENT input is stable (will not accept 2 successive changes of state) Otherwise, the DFB indicates an error. If ENABLE = 0, the EVENT input is not monitored. The default value of EVENT is 0.
COND	bit	R (1)	Input bit to be monitored for 1, regardless of the state of the ENABLE input. If the DFB is executed and if this bit changes to 0, the DFB indicates an error. The default value of COND1 is 1.

(1) : Access by program

Output parameters

Parameter	Type	Access	Description
ERROR	bit	R (1)	Error bit. This bit is set to 1 as soon as an error occurs. This bit is set to 0 if the ED input returns to 0 or if no errors occur.
STATUS	word	R (1)	This word indicates the type of error (see section 2.4). This word is at 0 if there are no errors. This word is set to 0 if the ED input returns to 0 or if no errors occur.

(1) : Access by program

Public variables

Parameter	Type	Access	Description
VALUE	bit	R/W (1)	The value (0 or 1) with which the EVENT input is compared The default value of VALUE is 1.
AREA_NR	word	R (1)	This word is used to specify which area of the control system is monitored by the diagnostic DFB. Example : Machining : no.1 Milling : no. 2 Tapping : no. 3 AREA_ NR must have the value 1, 2 or 3 so that the user can identify the faulty part of the control system. The above areas should be divided in the same way as the function modules. AREA_ NR can take a value from 0 to 15 (the default value is 0).
OP_CTRL	bit	R (1)	This bit indicates whether or not acknowledgment of the DFB instance by the operator is required. OP_CTRL = 0 : no acknowledgment by the operator OP_CTRL = 1 : acknowledgment by the operator The default value of OP_CTRL is 0.

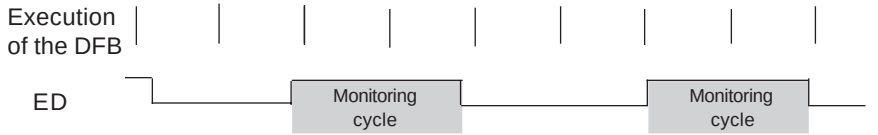
(1) : Access by program

2.4 List of Errors

STATUS word

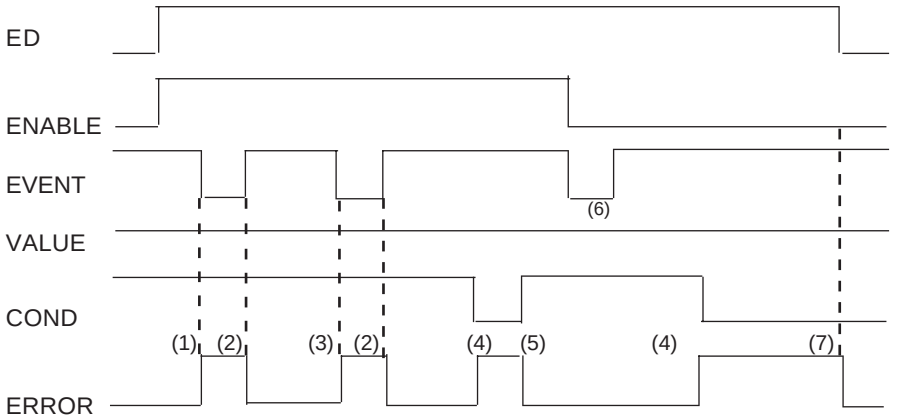
bit0 = 1 : EVENT different from the value specified in VALUE
bit1 = 1 : COND does not have the expected value 1
bit2 : Not significant
to
bit7 : Not significant
bit8 = 1 : EVENT unstable
bit9 : Not significant
to
bit15 : Not significant

2.5 Operation



Each time the DFB is executed, it performs the following processing operations :

- Input acquisition (ED, ENABLE, EVENT, COND)
- Input monitoring
- Output updating (ERROR, STATUS)



(1) An error is detected when the EVENT input is different from the VALUE public variable (ENABLE = 1).

(2) The ERROR output changes to zero when the EVENT input takes the value of the VALUE public variable.

(3) An error is detected when the EVENT input becomes unstable.

(4) An error is detected when the COND input is different from 1.

(5) The ERROR output changes to zero when the COND input takes the value 1.

(6) The EVENT input is different from the VALUE public variable : no errors are present because the ENABLE input is at 0.

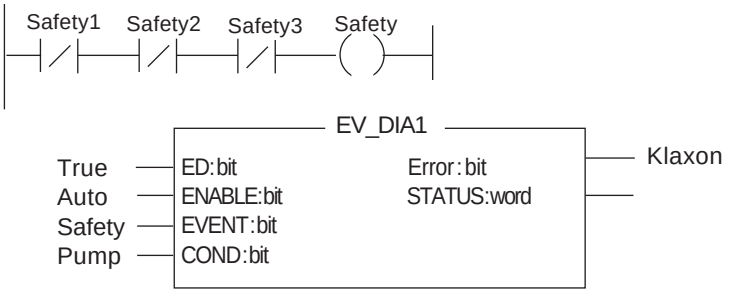
(7) The ERROR output changes to zero when the ED input takes the value 0.

For example : Diagnostics for activation of automatic safety interlocks

Checks to perform

- In automatic operation, check that the safety interlock systems are not triggered.
- Continually check that a hydraulic pump remains under pressure.

Principle of the PL7 program



Representation of the DFB in ST language

%L0 :
EV_DIA1 (True, Auto, Safety, Pump, Klaxon,);

2.5-1 Behavior of the DFB on Error Detection

When one of the monitored inputs is no longer in the state configured for the DFB, the DFB indicates an error when updating the following outputs :

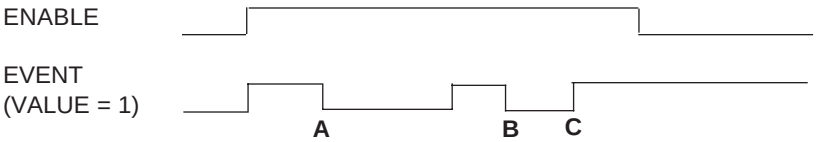
- ERROR bit set to 1
- STATUS word bit corresponding to the error set to 1

All errors detected during the same monitoring cycle are retained as they occur (the STATUS word bit corresponding to output updating is set to 1).

At the end of a monitoring cycle (falling edge on the ED input), the ERROR and STATUS outputs are reset to 0.

"EVENT input unstable" error

This error is caused by 2 changes in state of the EVENT input during the same monitoring cycle.



Error A : The EVENT input no longer has the value specified in VALUE.
Error B : The EVENT input is unstable.

The "EVENT input unstable" error changes to an "EVENT is different from VALUE" error if there are more than 1000 PLC scans between B and C.

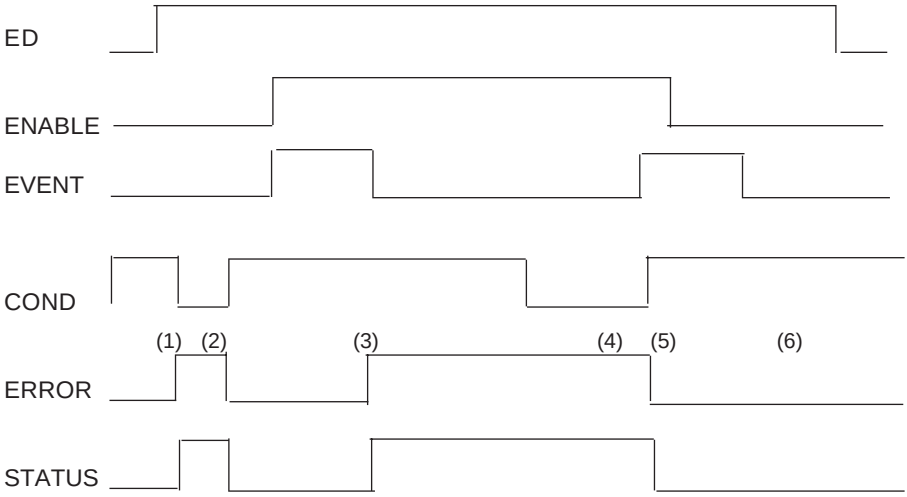
The "EVENT input unstable" error disappears after C if the number of PLC scans is greater than 1000, and if the EVENT input is still equal to the value specified in VALUE.

2.5-2 Behavior of the DFB on a Power Break

On a cold restart, the DFB initializes its parameters and public variables as follows :

- The COND input is set to 1 and the other inputs are set to 0.
- The outputs are set to 0.
- VALUE is set to 1.

2.5-3 Operating Example



(1) The COND bit is different from 1. The ERROR bit changes to 1 and the STATUS word indicates this error (bit 1 = 1).

(2) On a rising edge of the COND bit, the ERROR bit and bit 1 of the STATUS word change to 0.

(3) The EVENT bit is different from the value specified in VALUE (= 1 by default). The ERROR bit changes to 1 and the STATUS word indicates the error (bit 0 at 1).

(4) The COND bit is different from 1. The ERROR bit does not change and bit 1 of the STATUS word changes to 1.

(5) The EVENT bit is equal to the value specified in VALUE and the COND bit is at 1. The ERROR bit and the STATUS word change to 0.

(6) The EVENT bit is different from the value specified in VALUE (= 1 by default). The ERROR bit remains at 0 and the STATUS word does not indicate this error because the ENABLE input is at 0 (event monitoring).

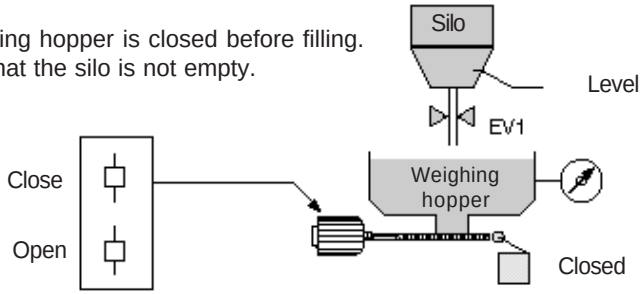
2.6 Example of Use

Checking hopper filling

Cycle : drop 100 kg of product into the weighing hopper

Checks to perform

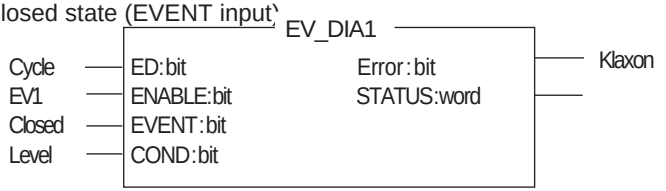
- Check that the weighing hopper is closed before filling.
- Continuously check that the silo is not empty.



PL7 program

```
%L0:
EV_DIA1 (Cycle, EV1, Closed; Level, Klaxon);
(*Filling hopper*)
! IF (Cycle AND Closed)
THEN
  SET EV1;
ELSE
  RESET EV1;
END_IF;
(*Hopper door control*)
! IF Weight >= 100
THEN
  RESET EV1;
  RESET Close;
  SET Open;
END_IF;
! IF Weight = 0
THEN
  RESET Open;
  SET Close;
END_IF;
```

- The level of the product in the silo is checked continuously while the cycle is in progress.
- When the weighing hopper is full (EV1 on ENABLE), the hopper door is monitored for the Closed state (EVENT input).

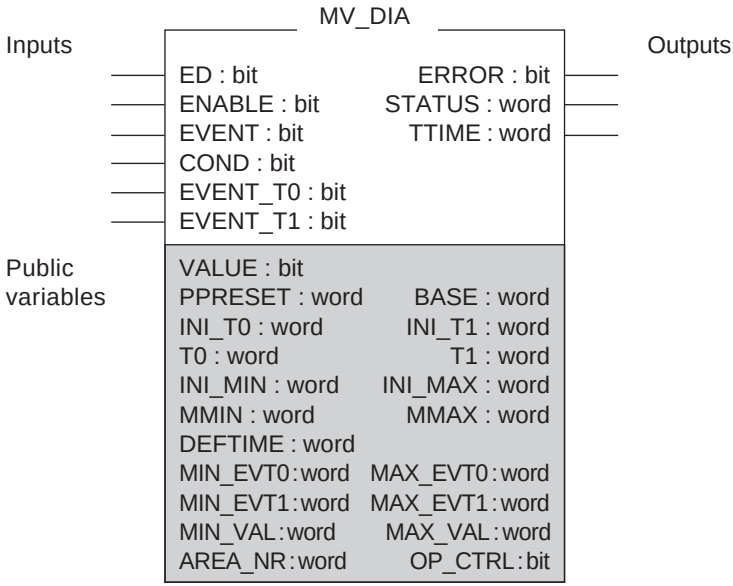


3.1 General

This DFB is used to monitor :

- The state of a bit without taking a time factor into account
- A movement (change of state of a bit within a defined time period)

3.2 Presentation of MV_DIA



3.3 Description of the Parameters

Input parameters

Parameter	Type	Access	Description
ED	bit	R (1)	DFB activation bit If ED = 0, the EVENT, EVENT_T0, EVENT_T1 and COND inputs are not monitored. The default value of ED is 0.
ENABLE	bit	R (1)	Monitoring enable bit : If ENABLE = 0, only the COND input is monitored. If ENABLE = 1, the COND, EVENT, EVENT_T0, EVENT_T1 inputs are all monitored. The default value of ENABLE is 0.
EVENT	bit	R (1)	Input bit to be monitored. If the DFB is executed and if ENABLE = 1, the DFB monitors that : • The EVENT input takes the value specified in the VALUE internal data variable • The EVENT input is stable (will not accept 2 successive changes of state) • The EVENT input takes the value specified in VALUE, with a minimum time MMIN and a maximum time MMAX Otherwise, the DFB indicates an error. The default value of EVENT is 0.
COND	bit	R (1)	Input bit to be monitored for 1. If the DFB is executed and if this bit changes to 0, the DFB indicates an error. The default value of COND1 is 1.
EVENT_T0	bit	R (1)	External event associated with time T0 This optional parameter is a bit which must change from 0 to 1 before time T0 or while ENABLE remains at 1. The default value of EVENT_T0 is 1.
EVENT_T1	bit	R (1)	External event associated with time T1. This optional parameter is a bit which must change from 0 to 1 before time T1 or while ENABLE remains at 1. The default value of EVENT_T1 is 1.

(1) : Access by program

Output parameters

Parameter	Type	Access	Description
ERROR	bit	R (1)	Error bit. This bit is set to 1 as soon as an error occurs. This bit is set to 0 if the ED input returns to 0 or if the error disappears.
STATUS	word	R (1)	This word indicates the type of error (see section 3.4).
TTIME	word	R (1)	This word indicates the current time in a time base expressed as multiples of N x 100 ms. Factor N is defined by the BASE public variable. TTIME is initialized to the PPRESET value and starts to evolve on a rising edge of the ENABLE input. It stops evolving and is frozen in this state on a falling edge of ENABLE. If an error is detected (ERROR = 1), TTIME is frozen in this state until ERROR returns to 0, then • If ENABLE = 0, TTIME = 0 • If ENABLE = 1, TTIME = current internal time value

(1) Access by program

Public variables

Parameter	Type	Access	Description
VALUE	bit	R/W (1)	The value (0 or 1) with which the EVENT input is compared. The default value of VALUE is 1.
PPRESET	word	R/W (1)	This word is used to define (by program or by modifying a variable) the initialization value of the current time (TTIME) on the rising edge of ENABLE. The default value of PPRESET is 0.
T0	word	R/W (1)	This word defines the maximum time T0 for the EVENT_T0 input to change from 0 to 1. If this change of state occurs after time T0, the DFB indicates an error. The default value of T0 is 0.
T1	word	R/W (1)	This word defines the maximum time T1 for the EVENT_T1 input to change from 0 to 1. If this change of state occurs after time T1, the DFB indicates an error. The default value of T1 is 0.
MMIN	word	R/W (1)	This word defines the minimum time that the EVENT input must remain equal to the VALUE internal data variables. As soon as the EVENT input differs from VALUE before time MMIN, the DFB indicates an error. If this error is the first one on the EVENT input since the last initialization (ENABLE 0 -> 1), the corresponding time (MMIN) is recorded in DEFTIME. The default value of MMIN is 0.
MMAX	word	R/W (1)	This word defines the maximum time that the EVENT input must remain equal to the VALUE internal data variable. If the EVENT input remains equal to VALUE after time MMAX, the DFB indicates an error. If this error is the first one on the EVENT input since the last initialization (ENABLE 0 -> 1), the corresponding time (MMAX) is recorded in DEFTIME.
DEFTIME	word	R/W (1)	This word records the time corresponding to the occurrence of the first error on the EVENT input. DEFTIME is initialized at 0 on a falling edge of the ED input. The default value of DEFTIME is 0.

(1) : Access by program

Public variables (continued)

Parameter	Type	Access	Description
MIN_EVT0	word	R/W (1)	This word records the minimum time required for the EVENT_T0 input to change from 0 to 1. MIN_EVT0 is initialized at 32767 on a rising edge of the ED input. The default value of MIN_EVT0 is 32767.
MIN_EVT1	word	R/W (1)	This word records the minimum time required for the EVENT_T1 input to change from 0 to 1. MIN_EVT1 is initialized at 32767 on a rising edge of the ED input. The default value of MIN_EVT1 is 32767.
MAX_EVT0	word	R/W (1)	This word records the maximum time required for the EVENT_T0 input to change from 0 to 1. MAX_EVT0 is initialized at 0 on a rising edge of the ED input. The default value of MAX_EVT0 is 0.
MAX_EVT1	word	R/W (1)	This word records the maximum time required for the EVENT_T1 input to change from 0 to 1. MAX_EVT1 is initialized at 0 on a rising edge of the ED input. The default value of MAX_EVT1 is 0.
MIN_VAL	word	R/W (1)	This word records the minimum time that the EVENT input retained the value specified in the VALUE data variable. MIN_VAL is reset to 32767 on a rising edge of the ED input. The default value of MIN_VAL is 32767.
MAX_VAL	word	R/W (1)	This word records the maximum time that the EVENT input retained the value specified in the VALUE data variable. MAX_VAL is reset to 0 on a rising edge of the ED input. The default value of MAX_VAL is 0.
INI_T0	word	R (1)	This word indicates the initial value of T0. This value is transferred to T0 on start-up or on a cold restart. The default value of INI_T0 is 0.
INI_T1	word	R (1)	This word indicates the initial value of T1. This value is transferred to T1 on start-up or on a cold restart. The default value of INI_T1 is 0.

(1) : Access by program

Public variables (continued)

Parameter	Type	Access	Description
INI_MIN	word	R (1)	This word indicates the initial value of time MMIN. This value is transferred to MMIN on start-up or on a cold restart. The default value of INI_MIN is 0.
INI_MAX	word	R (1)	This word indicates the initial value of time MMAX. This value is transferred to MMAX on start-up or on a cold restart. The default value of INI_MAX is 0.
BASE	word	R (1)	This word defines the factor N required for defining the time base. All times are expressed as multiples of N x 100 ms. The default value of BASE is 1.
AREA_NR	word	R (1)	This word is used to specify which area of the control system is monitored by the diagnostic DFB. Example : Machining : no.1 Milling : no. 2 Tapping : no. 3 AREA_ NR must have the value 1, 2 or 3 so that the user can identify the faulty part of the control system. The above areas should be divided in the same way as the function modules. AREA_ NR can take a value from 0 to 15 (the default value is 0).
OP_CTRL	bit	R (1)	This bit indicates whether or not acknowledgment of the DFB instance by the operator is required. OP_CTRL = 0 : no acknowledgment by the operator OP_CTRL = 1 : acknowledgment by the operator The default value of OP_CTRL is 0.

(1) : Access by program

3.4 List of Errors

STATUS word

bit 0 = 1 : EVENT observed different from the value specified in VALUE

bit 1 = 1 : COND does not have the expected value 1

bit 2 = 1 : EVENT has not had the value specified in VALUE over the entire specified MIN duration

bit 3 = 1 : EVENT has had the value specified in VALUE beyond the specified MAX duration

bit 4 = 1 : EVENT-T0 not seen at 1 before the specified time T0

bit 5 = 1 : EVENT-T1 not seen at 1 before the specified time T1

bit 6 = 1 : EVENT-T0 not seen at 1 while ENABLE = 1

bit 7 = 1 : EVENT-T1 not seen at 1 while ENABLE = 1

bit 8 = 1 : EVENT unstable

bit 9 = 1 : EVENT-T0 returned to 0 after time T0

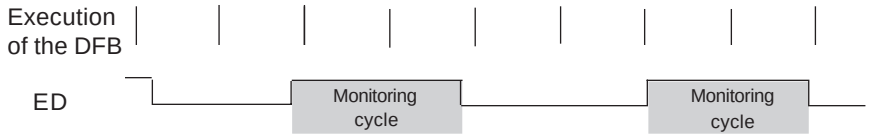
bit 10 = 1 : EVENT-T1 returned to 0 after time T1

bit 11 : Not significant
to
bit 13 : Not significant

bit 14 = 1 : Internal clock overrun error

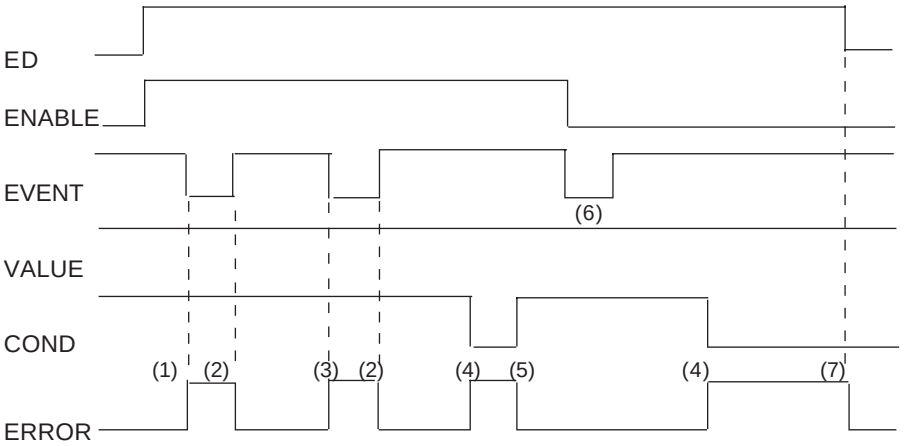
bit 15 : Not significant

3.5 Operation



Each time the DFB is executed, it performs the following processing operations :

- Input acquisition (ED, ENABLE, EVENT, EVENT_T0, EVENT_T1, COND)
- Input monitoring
- Output updating (ERROR, STATUS)



(1) An error is detected when the EVENT input is different from the VALUE public variable (ENABLE = 1).

(2) The ERROR output changes to zero when the EVENT input takes the value of the VALUE public variable.

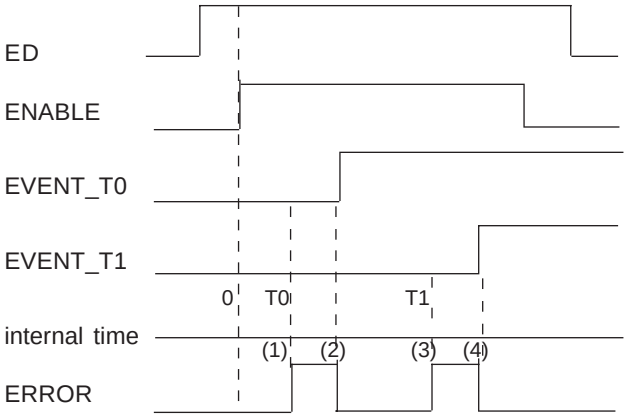
(3) An error is detected when the EVENT input becomes unstable.

(4) An error is detected when the COND input is different from 1.

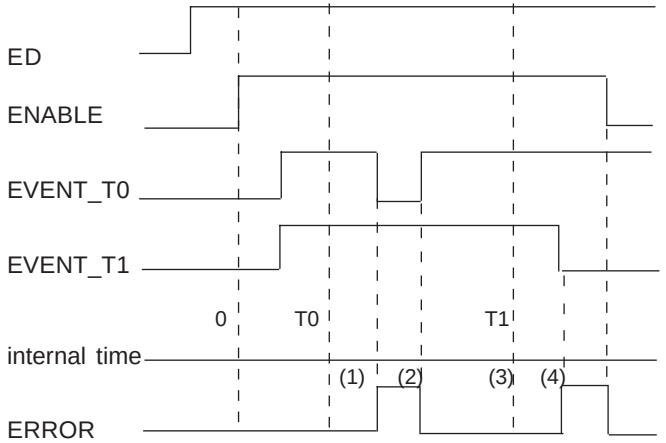
(5) The ERROR output changes to zero when the COND input takes the value 1.

(6) The EVENT input is different from the VALUE public variable : no errors are present because the ENABLE input is at 0.

(7) The ERROR output changes to zero when the ED input takes the value 0.



(1) An error is detected when the EVENT_T0 input has not changed to 1 before T0 times out.
(2) The ERROR output changes to zero when the EVENT_T0 input takes the value 1.
(3) An error is detected when the EVENT_T1 input has not changed to 1 before T1 times out.
(4) The ERROR output changes to zero when the EVENT_T1 input takes the value 1.



(1) An error is detected when the EVENT_T0 input has not stayed at 1 after T0 times out.
(2) The ERROR output changes to zero when the EVENT_T0 input takes the value 1.
(3) An error is detected when the EVENT_T1 input has not stayed at 1 after T1 times out.
(4) The ERROR output changes to zero when the ENABLE input changes to 0.

The time base used for T0, T1, MMIN and MMAX is defined by BASE. Any change in the value of BASE is not taken into account during the current monitoring cycle. It will be accepted at the start of the next cycle.

3.5-1 Behavior of the DFB on Error Detection

When one of the monitored inputs is no longer in the state configured for the DFB, the DFB indicates an error when updating the following outputs :

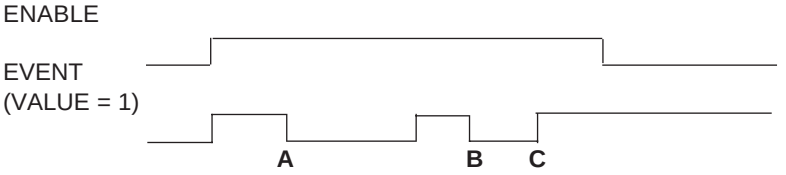
- ERROR bit set to 1
- STATUS word bit corresponding to the error set to 1

All errors detected during the same monitoring cycle are retained as they occur (the STATUS word bit corresponding to output updating is set to 1).

At the end of a monitoring cycle (falling edge on the ED input), the ERROR and STATUS outputs are reset to 0.

"EVENT input unstable" error

This error is caused by 2 changes in state of the EVENT input during the same monitoring cycle.



Error A : The EVENT input no longer has the value specified in VALUE.
Error B : The EVENT input is unstable.

The "EVENT input unstable" error changes to an "EVENT is different from VALUE" error if there are more than 1000 PLC scans between B and C.

The "EVENT input unstable" error disappears after C if the number of PLC scans is greater than 1000, and if the EVENT input is still equal to the value specified in VALUE.

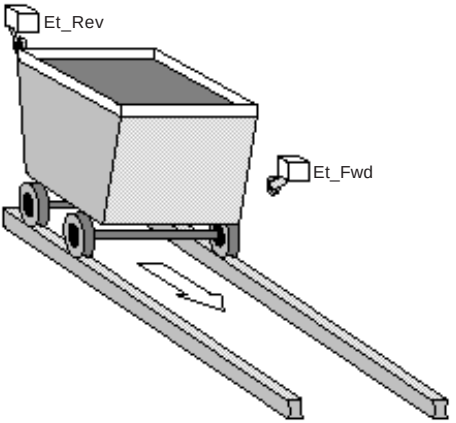
3.5-2 Behavior of the DFB on a Power Break

On a cold restart, the DFB initializes its parameters and public variables as follows :

- The COND, EVENT_T0 and EVENT_T1 inputs are set to 1.
- The other inputs (ENABLE, EVENT) are set to 0.
- The ERROR, STATUS and TTIME outputs are set to 0.
- VALUE is set to 1.
- INI_T0, INI_T1, INI_MIN and INI_MAX are transferred to T0, T1, MMIN and MMAX respectively.
- MIN_EVT0, MIN_EVT1 and MIN_VAL are set to 32767.
- The other data variables (PPRESET, DEFTIME, MAX_EVT0, MAX_EVT1 and MAX_VAL) are set to 0.

3.6 Example of use

Checking the movement of a cart



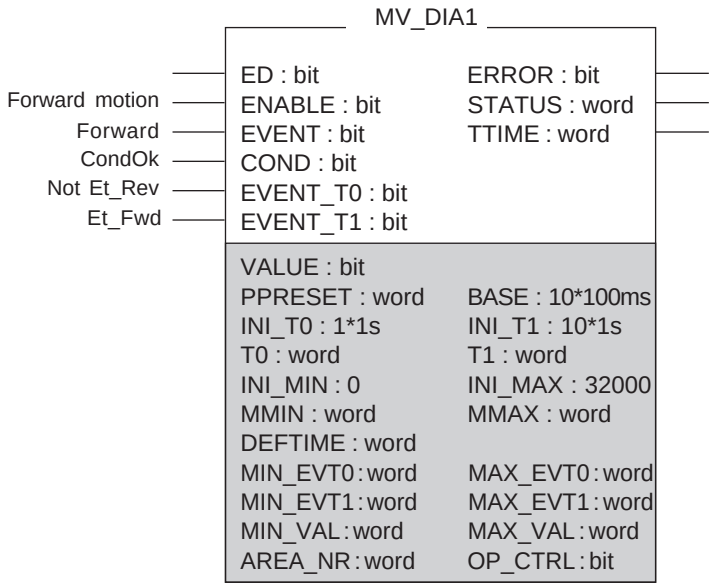
Checks to perform

- Check that the Forward motion command was sent.
- After reception of the Forward motion command, check that the cart moves away from the Et_Rev sensor in less than 1 second.
- Check that the duration of Forward motion does not exceed 10 seconds.
- Check that the 2 end of travel sensors are never at 1 simultaneously.
- Check that the Et_Rev sensor is at 1 when the cart is stopped.

PL7 program

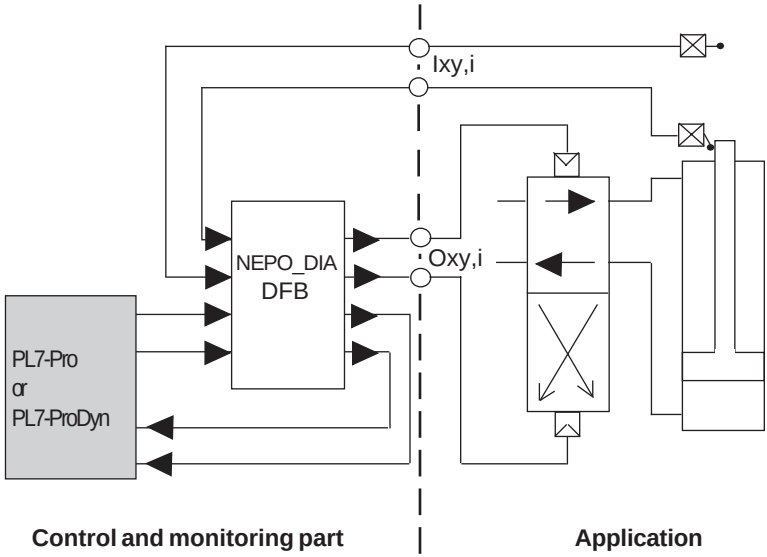
%L0:
Forward motion := Forward AND NOT Et_Fwd;
CondOK := Not (Et_Fwd AND Et_Rev) AND (Et_Fwd OR Forward motion OR Et_Fwd) ;
MV_DIA1 (Forward motion, Forward, CondOK, Not Et_Rev, Et_Fwd, , ,) ;

- The EVENT input is used to check while the cart moves that the Forward command was given correctly.
- The EVENT_T0 input is used to check that the cart moves away from the Et_Rev sensor in less than 1 second.
- The EVENT_T1 input is used to check that Forward motion does not continue for more than 10 seconds.
- The COND input is monitored for state 1 for the entire duration of DFB execution. This ensures that :
  - The Et_Rev sensor is at 1 when the cart is stopped.
  - The Et_Rev and Et_Fwd sensors are never at 1 simultaneously.


MV_DIA1
Forward motion
Forward
CondOk
Not Et_Rev
Et_Fwd
ED : bit
ENABLE : bit
EVENT : bit
COND : bit
EVENT_T0 : bit
EVENT_T1 : bit
ERROR : bit
STATUS : word
TTIME : word
VALUE : bit
PPRESET : word
BASE : 10*100ms
INI_T0 : 1*1s
INI_T1 : 10*1s
T0 : word
T1 : word
INI_MIN : 0
INI_MAX : 32000
MMIN : word
MMAX : word
DEFTIME : word
MIN_EVT0 : word
MAX_EVT0 : word
MIN_EVT1 : word
MAX_EVT1 : word
MIN_VAL : word
MAX_VAL : word
AREA_NR : word
OP_CTRL : bit

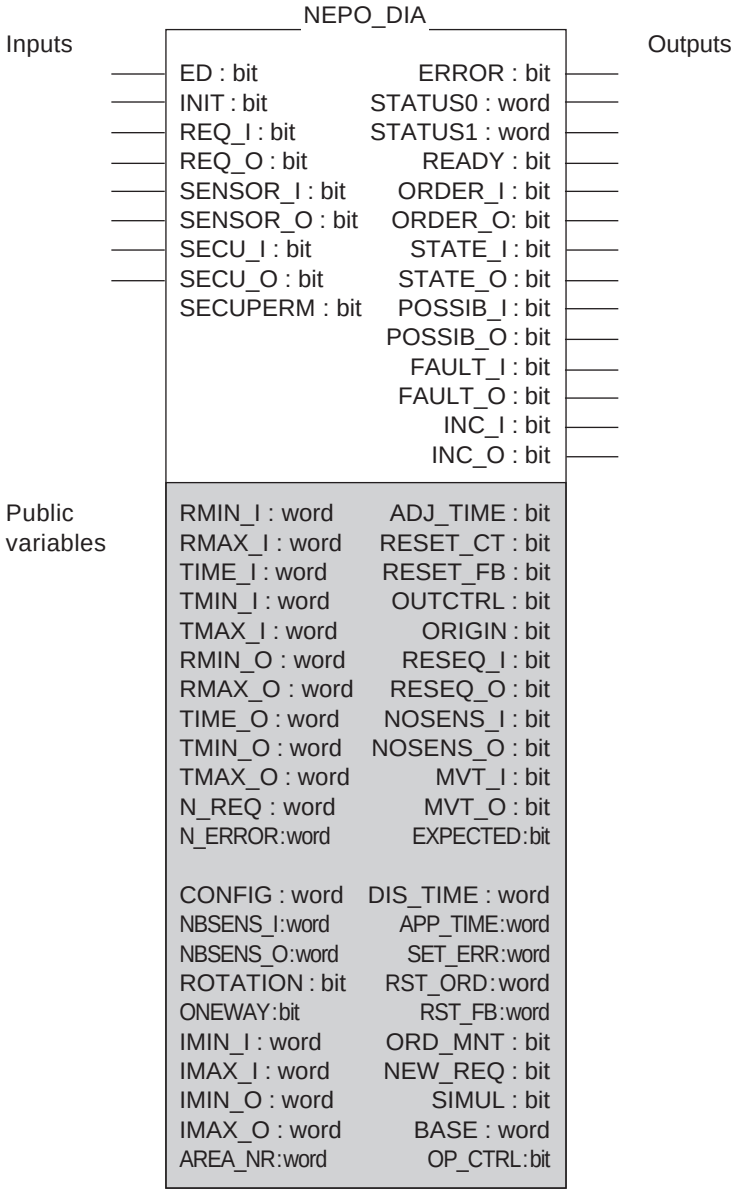
4.1 General

These DFBs are used to monitor, control and diagnose an **application component (EPO)**, ie. a device directly affecting the manufacturing process and the environment. For this, the DFB, defined by a "preactuator-actuator/sensor" combination, ensures positioning between two reference points (which may or may not be monitored), with movement (linear or rotating) at a constant speed. This concerns the control of cylinders (monostable or bistable cylinders or directional control valves) or the control of certain motors used for positioning, clamping, machining, turntables, etc.



Note
The TEPO_DIA DFB is exactly the same as the NEPO_DIA DFB. Its only limitation is that it can only manage linear movement (no rotation).
As a result, the ROTATION and ONEWAY public variables do not exist for this DFB.

4.2 Presentation of NEPO_DIA



4.3 Description of the Parameters

Input parameters

Parameter	Type	Access	Description
ED	bit	R (1)	DFB activation bit If ED = 0, the DFB is not executed. The default value of ED is 0.
INIT	bit	R (1)	At state 1, this bit acknowledges the faults indicated by the ERROR bit and the STATUS0 word. It is reset to 0 by the DFB. The default value of INIT is 0.
REQ_I	bit	R (1)	This bit is set to 1 by the control part in order to request an "incoming" movement. The default value of REQ_I is 0.
REQ_O	bit	R (1)	This bit is set to 1 by the control part in order to request an "outgoing" movement. The default value of REQ_O is 0.
SENSOR_I	bit	R (1)	This input receives the position information from all the "incoming" position sensors. The default value of SENSOR_I is 0.
SENSOR_O	bit	R (1)	This input receives the position information from all the "outgoing" position sensors. The default value of SENSOR_O is 0.
SECU_I	bit	R (1)	This input enables the user to wire the safety conditions of an "incoming" movement. The default value of SECU_I is 0.
SECU_O	bit	R (1)	This input enables the user to wire the safety conditions of an "outgoing" movement. The default value of SECU_O is 0.
SECUPERM	bit	R (1)	This input enables the user to wire the permanent operating conditions. The default value of SECUPERM is 0.

(1) : Access by program

Output parameters

Parameter	Type	Access	Description
ERROR	bit	R (1)	Error bit. This bit is set to 1 as soon as an error occurs and provided that this error is not masked (see the SET_ERR public variable). The default value of ERROR is 0.
STATUS0 STATUS1	word	R (1)	These 2 words indicate the type of error. STATUS0 indicates the errors related to the DFB operation. STATUS1 is reserved for configuration errors (see section 4.6). The default value of STATUS0 is 0 and STATUS1 is 0.
READY	bit	R (1)	This bit indicates the availability of the DFB : at state 1, the DFB is in **command** mode (setting orders) and at state 0, the DFB is in **recalibration** mode (awaiting reference point). The default value of READY is 0.
ORDER_I	bit	R (1)	At state 1, this bit indicates that the "incoming" command is activated. The default value of ORDER_I is 0.
ORDER_O	bit	R (1)	At state 1, this bit indicates that the "outgoing" command is activated. The default value of ORDER_O is 0.
STATE_I	bit	R (1)	At state 1, this bit indicates that the "incoming" position is monitored. The default value of STATE_I is 0.
STATE_O	bit	R (1)	At state 1, this bit indicates that the "outgoing" position is monitored. The default value of STATE_O is 0.
POSSIB_I	bit	R (1)	This bit indicates that the DFB is ready to accept an "incoming" movement request. The default value of POSSIB_I is 0.
POSSIB_O	bit	R (1)	This bit indicates that the DFB is ready to accept an "outgoing" movement request. The default value of POSSIB_O is 0.

(1) : Access by program

Output parameters (continued)

Parameter	Type	Access	Description
FAULT_I	bit	R (1)	This bit indicates a fault detected during an "incoming" movement (out of position). The default value of FAULT_I is 0.
FAULT_O	bit	R (1)	This bit indicates a fault detected during an "outgoing" movement (out of position). The default value of FAULT_O is 0.
INC_I	bit	R (1)	This bit indicates (in the absence of orders or requests) an inconsistency between the "incoming" state expected by the control system (RESEQ_I or ORIGIN data variables) and the position known to the DFB. The default value of INC_I is 0.
INC_O	bit	R (1)	This bit indicates (in the absence of orders or requests) inconsistency between the "outgoing" state expected by the control system (RESEQ_O data variable) and the position known to the DFB. The default value of INC_O is 0.

(1) : Access by program

Public variables associated with time management

The values of these data variables give a time equal to n x 100 ms, where n is the value of the time base constant (BASE). The accepted values are whole numbers between 0 and 32767 inclusive.

Parameter	Type	Access	Description
RMIN_I, RMIN_O	word	R/W (1)	These 2 words serve as a reference for the minimum duration of incoming/outgoing movements respectively. By default or on a RESET_FB request, these words are set to the value of IMIN_I and IMIN_O respectively (or to 0 if IMIN_I = IMAX_I = 0, IMIN_O = IMAX_O = 0).
RMAX_I, RMAX_O	word	R/W (1)	These 2 words serve as a reference for the maximum duration of the incoming/outgoing movements respectively. By default or on a RESET_FB request, these words are set to the value of IMAX_I and IMAX_O respectively (or to 32767 if IMIN_I = IMAX_I = 0, IMIN_O = IMAX_O = 0).
TIME_I, TIME_O	word	R (1)	These 2 words contain the actual time of current incoming/outgoing movements respectively or the time of the last incoming/outgoing movement performed respectively. The default value of TIME_I is 0 and TIME_O is 0.
TMIN_I, TMIN_O	word	R (1)	These 2 words record the minimum time necessary for the incoming/outgoing movements respectively. By default or on request, RESET_CT, TMIN_I and TMIN_O take the value RMAX_I or RMAX_O if ADJ_TIME = 1, and IMAX_I or IMAX_O if ADJ_TIME = 0.
TMAX_I, TMAX_O	word	R (1)	These 2 words record the maximum time necessary for the incoming/outgoing movements respectively. By default or on request, RESET_CT, TMAX_I and TMAX_O take the value RMIN_I or RMIN_O if ADJ_TIME = 1, and IMIN_I or IMIN_O if ADJ_TIME = 0.

(1) : Access by program

Public variables used as reliability indicators

Parameter	Type	Access	Description
N_REQ	word	R (1)	This word records the number of requests accepted by the DFB. It takes the value 0, when RESET_CT is set to 1 or if the counter is full (when the limit of 32767 is reached). If the N_REQ counter overflows, it is reset to 0, together with the N_ERROR counter.
N_ERROR	word	R (1)	This word records the number of errors detected by the DFB (rising edges on the ERROR bit). It takes the value 0, when RESET_CT is set to 1 or if the counter is full (when the limit of 32767 is reached). If the N_ERROR counter overflows, it is reset to zero, together with the N_REQ counter.

Public variables used for specific requests

Parameter	Type	Access	Description
RESET_CT	bit	R/W (1)	At state 1, this bit reinitializes the counters which record the minimum, maximum and current times of the incoming/ outgoing movements (TMIN_I,TMIN_O, TMAX_I, TMAX_O, TIME_I and TIME_O), the number of movement requests accepted (N_REQ) and the number of errors detected (N_ERROR). It is reset to 0 by the DFB. The default value of RESET_CT is 0.
RESET_FB	bit	R/W (1)	At state 1, this bit reinitializes the DFB (except for the data variables managed by RESET_CT). It is reset to 0 by the DFB. The default value of RESET_FB is 0.

(1) : Access by program

Public variables used for restarting a cycle

Parameter	Type	Access	Description
OUTCTRL	bit	R/W (1)	After an error selected in RST_FB, this data variable authorizes the DFB to send orders without monitoring the sensors, in order to bring the application to a controlled position for recalibration. The SECU_I, SECU_O and SECUPERM inputs must be valid. The default value of OUTCTRL is 0.
ORIGIN	bit	R/W (1)	This bit indicates that the "reference point" position is expected by the control system (equivalent to RESEQ_I but taking priority). The default value of ORIGIN is 0.
RESEQ_I	bit	R/W (1)	This bit indicates that the control system is expecting the "incoming" status. The default value of RESEQ_I is 0.
RESEQ_O	bit	R/W (1)	This bit indicates that the control system is expecting the "outgoing" status. The default value of RESEQ_O is 0.

Public variables used for position monitoring

Parameter	Type	Access	Description
NOSENS_I, NOSENS_O	bit	R/W (1)	These bits have the opposite states to the sensors wired on the respective SENSOR_I and SENSOR_O inputs. These bits are only used if the DFB is configured to monitor the positions using this data (internal constants NBSENS_I and/or NBSENS_O at 2).

Public variables indicating status

Parameter	Type	Access	Description
ADJ_TIME	bit	R (1)	This bit indicates that the movement reference times have been acquired (learning mode). The default value of ADJ_TIME is 0.
MVT_I	bit	R (1)	These 2 bits indicate the transitory state of an MVT_O incoming/outgoing movement started but not finished (target position not reached). The default value of MVT_I is 0 and MVT_O is 0.
EXPECTED	bit	R (1)	This bit indicates that the DFB is waiting for the appearance of an end of movement sensor (movement started more than RMIN_I or RMIN_O ago or was stopped). The default value of EXPECTED is 0.

(1) : Access by program

Public variables used for configuring the types of actuators controlled

Parameter	Type	Access	Description
CONFIG	word	R (1)	This word configures the type of actuator controlled (see section 4.5). The default value of CONFIG is -1 (this value is deliberately out of range to force selection of the type of actuator to use).
NBSENS_I, NBSENS_O	word	R (1)	These 2 words define the way in which the DFB monitors the incoming/outgoing positions respectively: • NBSENS_I (or NBSENS_O) = 0 : position is not monitored • NBSENS_I (or NBSENS_O) = 1 : position is monitored by the SENSOR_I (or SENSOR_O) input • NBSENS_I (or NBSENS_O) = 2 : position is monitored by the SENSOR_I (or SENSOR_O) input (on state of all the sensors) and the NOSENS_I (or NOSENS_O) public variable (off state of all the sensors). The default value of NBSENS_I is 1 and NBSENS_O is 1.
ROTATION	bit	R (1) (2)	At state 1 this bit defines a rotating movement. The default value of ROTATION is 0 (linear movement).
ONEWAY	bit	R (1) (2)	At state 1 this bit defines a rotating movement, with the possibility of linking several movements in the same direction. The default value of ONEWAY is 0.
SIMUL	bit	R (1)	At state 1, this bit sets the DFB to simulation mode. The default value of SIMUL is 0.

Public variables associated with time management

The values of these constants give a time equal to n x 100 ms, where n is the value of the time base constant (BASE). The accepted values are whole numbers between 0 and 32767 inclusive.

Parameter	Type	Access	Description
IMIN_I, IMIN_O	word	R (1)	These 2 words define the minimum time permitted for the incoming/outgoing movements respectively. When the DFB is initialized, the IMIN_I and IMIN_O values are copied to RMIN_I and RMIN_O respectively (if IMIN_I and IMIN_O are not both at 0). The default value of IMIN_I is 0 and IMIN_O is 0.
IMAX_I, IMAX_O	word	R (1)	These 2 words define the maximum time permitted for the incoming/outgoing movements respectively. When the DFB is initialized, the IMAX_I and IMAX_O values are copied to RMAX_I and RMAX_O respectively (if IMAX_I and IMAX_O are not both at 0). The default value of IMAX_I is 0 and IMAX_O is 0.

(1) : Access by program

(2) : These parameters do not exist for the TEPO_DIA DFB (no rotation).

Public variables associated with time management (continued)

Parameter	Type	Access	Description
DIS_TIME	word	R (1)	This word defines the duration during which the disappearance of a position sensor is accepted. The default value of DIS_TIME is 0.
APP_TIME	word	R (1)	This word defines the duration during which the unexpected appearance of a position sensor is accepted. The default value of APP_TIME is 0.
BASE	word	R (1)	This word represents the N factor needed to define the time base. All times are expressed as multiples of N x 100 ms. The default value of BASE is 1.

Public variables used for configuring the behavior of the DFB on an error

Parameter	Type	Access	Description
SET_ERR	word	R (1)	This word selects the errors causing the ERROR bit to be set to 1. The default value of SET_ERR is H'0FE7' (see section 4.4).
RST_ORD	word	R (1)	This word selects the errors causing orders to be reset to zero (ORDER_I and ORDER_O). These errors are recorded in STATUS0 until they have been acknowledged. They must also be selected in the SET_ERR mask. The default value of RST_ORD is H'0F87' (see section 4.4).
RST_FB	word	R (1)	This word selects the errors which switch the DFB to recalibration mode. These errors are recorded in STATUS0 until they have been acknowledged. They must also be selected in the SET_ERR mask. The default value of RST_FB is H'0187' (see section 4.4).

(1) : Access by program

Public variables used for configuring the DFB on restarting a cycle

Parameter	Type	Access	Description
ORD_MNT	bit	R (1)	If this bit is at state 1, the orders will be reactivated when STATUS0 ceases to indicate error(s) which reset the orders to zero. The default value of ORD_MNT is 0.
NEW_REQ	bit	R (1)	If this bit is at state 1, new requests will be required after an error is detected causing the DFB to change to recalibration mode (ie. an error selected in RST_FB). The default value of NEW_REQ is 1.

(1) : Access by program

Public variables used for the man-machine interface

Parameter	Type	Access	Description
AREA_NR	word	R (1)	This word is used to specify which area of the control system is monitored by the diagnostic DFB. Example : Machining : no.1 Milling : no. 2 Tapping : no. 3 AREA_ NR must have the value 1, 2 or 3 so that the user can identify the faulty part of the control system. The above areas should be divided in the same way as the function modules. AREA_ NR can take a value from 0 to 15 (the default value is 0).
OP_CTRL	bit	R (1)	This bit indicates whether or not acknowledgment of the DFB instance by the operator is required. OP_CTRL = 0 : no acknowledgment by the operator OP_CTRL = 1 : acknowledgment by the operator The default value of OP_CTRL is 0.

(1) : Access by program

4.4 Selection Masks for the Public Variables

The following table gives the default selection mask values for the SET_ERR, RST_ORD and RST_FB variables. When a bit is marked with a cross, this means that it is selected and that the corresponding error will not be masked. Thus the DFB allows a movement to be executed when an error is present. Eg. if bit 9, which selects the "disappearance of the permanent operating conditions" error, is at 0, orders can be activated even if this condition disappears.

Bit	Meaning	SET_ERR (H'0FE7')	RST_ORD (H'0F87')	RST_FB (H'0187')
0	Command error	x	x	x
1	"Incoming" sensor not expected	x	x	x
2	"Outgoing" sensor not expected	x	x	x
3	"Incoming" sensor untimely	.	.	.
4	"Outgoing" sensor untimely	.	.	.
5	"Incoming" sensor delayed	.	.	
6	"Outgoing" sensor delayed	.	.	
7	Disappearance of "incoming" sensor	x	x	x
8	Disappearance of "outgoing" sensor	x	x	x
9	Disapp. of permanent condition	x	x	.
10	Disapp. of "incoming" safety cond.	x	x	.
11	Disapp. of "outgoing" safety cond.	x	x	.
12	"Incoming" request refused	.	.	.
13	"Outgoing" request refused	.	.	.
14	"Incoming" sensor not deactivated	.	.	.
15	"Outgoing" sensor not deactivated	.	.	.

4.5 Selection of Actuators

It is the value of the CONFIG internal constant which selects the required type of actuator and order. The different configurations available are as follows :

Config	Actuator	Command	Command logic
0	monostable actuator, one order (ORDER_O)	one request (REQ_O)	order if requested (type 1)
1	monostable actuator, one order (ORDER_O)	two requests (REQ_O, REQ_I)	order maintained until opposite request (type 2)
2	monostable actuator, one order (ORDER_O)	two requests (REQ_O, REQ_I)	order if requested and latching of the order on the position, unlocked by opposite request or loss of position (type 5)
3	bistable actuator, two separate orders (ORDER_O, ORDER_I)	two requests (REQ_O, REQ_I)	order if requested (type 1)1
4	bistable actuator, two separate orders (ORDER_O, ORDER_I)	two requests (REQ_O, REQ_I)	order maintained until opposite request (type 2)
5	bistable actuator, two separate orders (ORDER_O, ORDER_I)	two requests (REQ_O, REQ_I)	order if requested and position not reached (type 3). The preactuator reacts on a pulse, pointless to maintain the order.
6	bistable actuator, two separate orders (ORDER_O, ORDER_I)	two requests (REQ_O, REQ_I)	order maintained until opposite request and position reached (type 4)
7	bistable actuator, two separate orders (ORDER_O, ORDER_I)	two requests (REQ_O, REQ_I)	order if requested and latching of the order on the position, unlocked by opposite request or loss of position (type 5)
8	multi-stable actuator, two separate orders (ORDER_O, ORDER_I)	two requests (REQ_O, REQ_I)	*same as 4*
9	multi-stable actuator	two requests (REQ_O, REQ_I)	*same as 6*
10	multi-stable actuator	two requests (REQ_O, REQ_I) and request missing	*same as 5* Intermediate stop authorized (requests missing)
11	multi-stable actuator	two requests (REQ_O, REQ_I) and request missing	*same as 7* Intermediate stop authorized (requests missing)

Note

CONFIG = 8 to 11 : Intermediate stop possible on error selected in RST_ORD.

4.6 List of Errors

When the DFB detects an error, this is indicated by the STATUS0 and STATUS 1 words (several errors can be indicated at once).

Whether or not the errors are recorded depends on the value of the masks for selecting the behavior of the DFB when there is an error : RST_ORD and RST_FB :

- An error selected in RST_FB will be recorded in STATUS0 until it disappears and is acknowledged by INIT (the DFB changes to recalibration mode).
- An error selected in RST_ORD will be recorded in STATUS0 until it disappears and is acknowledged by INIT (the DFB stays in control / monitoring mode).
- All other errors (not selected) are no longer indicated when the cause of the error disappears.

An error selected in SET_ERR sets the ERROR bit to 1.

The following table lists the types of errors indicated by the DFB :

Bit	Meaning of STATUS0 word
bit 0 = 1	**Error in the commands or irregular sensor data** The DFB has detected an irregular command or inconsistent position data. Irregular commands : "incoming" and "outgoing" movements occurring at the same time, use of the "incoming" command for a monostable actuator with a single request, "incoming" (RESEQ_I) and "outgoing" (RESEQ_O) states expected at the same time. Inconsistent position data : position sensors not acting together for a rotating movement, position not monitored and position sensor active, position monitored by several sensors and SENSOR_I / O and NOSENS_I / O variables active simultaneously.
bit 1 = 1 bit 2 = 1	**"Incoming" sensor not expected** **"Outgoing" sensor not expected** For position control, at least one opposite position sensor is active for longer than the time permitted and configured in APP_TIME. When deactivated, the sensor for the last position left is present again for longer than the time permitted and configured in APP_TIME. In recalibration mode, at least one sensor is present at each position.
bit 3 = 1 bit 4 = 1	**"Incoming" sensor untimely** **"Outgoing" sensor untimely** At least one sensor for the position to be reached is present before the minimum time defined for the movement in RMIN_I or RMIN_O.

List of errors (continued)

Bit	Meaning
bit 5 = 1 bit 6 = 1	**"Incoming" sensor delayed** **"Outgoing" sensor delayed** At least one sensor for the position to be reached is not yet present after the maximum time allocated to the movement and defined in RMAX_I or RMAX_O.
bit 7 = 1 bit 8 = 1	**Disappearance of the "incoming" sensor** **Disappearance of the "outgoing" sensor** For position control, at least one sensor has disappeared for longer than the time permitted and configured in DIS_TIME. In recalibration mode, no position is found again.
bit 9 = 1	**Disappearance of permanent condition** The permanent conditions have disappeared during a movement.
bit 10 = 1 bit 11 = 1	**Disappearance of safety condition for the "incoming" movement** **Disappearance of safety condition for the "outgoing" movement** The safety condition has disappeared during a movement.
bit 12 = 1 bit 13 = 1	**"Incoming" request refused** **"Outgoing" request refused** A request cannot be accepted by the DFB (safety conditions and/or permanent conditions missing, etc).
bit 14 = 1 bit 15 = 1	**"Incoming" sensor not deactivated** **"Outgoing" sensor not deactivated** At least one sensor for the last position left has not deactivated after the minimum time defined for the motion in RMIN_I or RMIN_O.

Configuration faults

When the DFB is initialized (application transfer, change of cartridge, etc), it has no context for the application and waits for a reference point. At this point, it can detect the configuration errors which are preventing it from operating. It indicates them via the output parameter STATUS1. The faults indicated are as follows :

Bit	Meaning of STATUS1 word
bit 0 = 1	Invalid type of actuator (incorrect CONFIG value)
bit 1 = 1	"Incoming" AND "outgoing" positions selected not monitored
bit 2 = 1	Rotating movement AND one of the positions selected not monitored
bit 3 = 1	Rotating movement, monostable and one-way
bit 4 = 1	Maximum duration for a movement less than or equal to the minimum duration
bit 5 = 1	Simulation and learning mode for the movement times
bit 6 = 1	One-way translational movement
bit 7 = 1	Learning mode for movement times and positions not monitored
bit 8 = 1	Rotating movement and positions monitored differently
bit 9 = 1	Selected CONFIG and selection mask RST_ORD incompatible
bit 10 = 1	Selected CONFIG AND position not monitored incompatible (actuator type 2, 7 or 11 and NBSENS_I or NBSENS_O = 0)
bit 11 = 1	Selection masks RST_ORD and RST_FB incompatible (the errors selected in RST_FB must also be selected in RST_ORD)
bit 12 = 1	Selection masks RST_ORD, RST_FB and SET_ERR incompatible (the errors selected in RST_FB et RST_ORD must also be selected in SET_ERR)
bit 13 = 1	Rotating movement AND selection mask RST_FB incompatible (ROTATION = 1 and sensor(s) not deactivated error not selected in RST_FB)

4.7 Operation

The DFB is inserted in the command providing the link between the application program and the action and vice versa : the REQ_O and REQ_I inputs receive requests and the ORDER_O and ORDER_I outputs transmit the orders to the actuator. The SENSOR_O and SENSOR_I inputs and the NOSENS_O and NOSENS_I data variables (if used) inform the DFB of the physical incoming/outgoing positions.

The movement time is monitored via the RMIN_O, RMAX_O, RMIN_I and RMAX_I data variables. The SECU_O and SECU_I inputs define the safety conditions which must be valid during the incoming/outgoing movements. The SECUPERM input represents the operating condition of the machine which must be valid during the movements.

4.7-1 Preprogramming the DFB

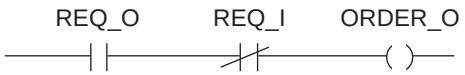
The user must preprogram the DFB in order to choose :

① The type of actuator controlled, defined by the CONFIG internal constant : monostable (ORDER_I not used) or bistable (ORDER_O and ORDER_I used).

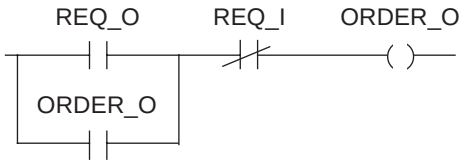
② The type of movement, defined by the ROTATION constant : translational or rotating. If the user chooses rotating movement, the incoming/outgoing position sensors act together and the ONEWAY constant defines whether the movement has either one or two rotation directions.

③ The type of orders given to the actuator. These orders are applied to the actuators according to the following equations for "outgoing" movements. These equations are identical for "incoming" movements (replace _O with _I and vice versa) :

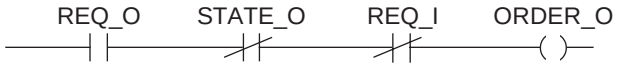
- Order if requested (type 1)



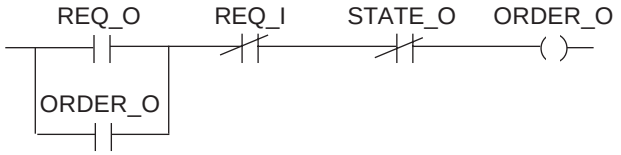
- Order recorded until opposite requested (type 2)



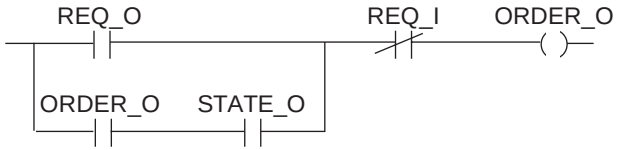
• Order if requested and until position reached (type 3)



• Order recorded until opposite requested and position reached (type 4)



• Order if requested and latching of the order at the position (type 5)



④ The way in which the physical positions of the application components (EPO) are monitored by the DFB. It is defined by the NBSENS_O and NBSENS_I internal constants.

NBSENS_O or NBSENS_I	**Monitoring**
0	Position not monitored. This position is considered as being reached if the DFB expects it to be reached, or not reached if the DFB does not expect it to be so. No fault linked to this position (sensor not deactivated, not expected, etc) will be indicated. In other words, this means that if a position is selected as not monitored, the DFB will stop the movement (to this position) as soon as the RMAX_I or RMAX_O time limit is reached and considers the application component as virtually in this position. Furthermore, on initialization or recalibration, the reference point can only be set to a monitored position.
1	Position monitored via the SENSOR_O or SENSOR_I input.
2	Position physically monitored using several sensors. The DFB monitors the position using 2 data variables : SENSOR_O (or SENSOR_I) and NOSENS_O (or NOSENS_I), where : POSITION_O = SENSOR_O . $\overline{\text{NOSENS\_O}}$ and POSITION_I = SENSOR_I . $\overline{\text{NOSENS\_I}}$ SENSOR_O or SENSOR_I represents the on state of all the sensors, NOSENS_O or NOSENS_I represents the off state of all the sensors.

Note

Both positions cannot be selected as not monitored at the same time. If so, the DFB indicates a configuration error (STATUS1) and becomes unusable.

⑤ The behavior of the DFB when an error is detected :

- The SET_ERR data variable defines the errors which will set the ERROR bit to 1.
- The RST_ORD data variable defines the errors which will deactivate the ORDER_I and ORDER_O outputs.
- The RST_FB data variable defines the errors which will switch the DFB to "recalibration" mode.

Setting a bit to 1 in one of the 2 data variables RST_ORD or RST_FB selects the error associated with the equivalent bit in STATUS0.

- The ORD_MNT data variable defines whether the orders must be reactivated or not on disappearance of indication in STATUS0 of errors which have caused orders to be reset during a motion.
- The NEW_REQ data variable defines whether new requests are required after an error occurs which sets the DFB to "recalibration" mode. New requests are required by default.

⑥ The movement times :

- The IMAX_I and IMAX_O data variables define the maximum times for the incoming/outgoing movements.
- The IMIN_I and IMIN_O data variables define the minimum times for the incoming/outgoing movements.

The values express times in N x 100 ms, where N is the value of the time base constant (BASE). When the DFB is initialized, these values are copied to the RMAX_I, RMAX_O, RMIN_I and RMIN_O data variables. If the IMIN_I and IMAX_I (or IMIN_O and IMAX_O) data variables, which define the movement time, are at zero, the DFB will learn the movement time.

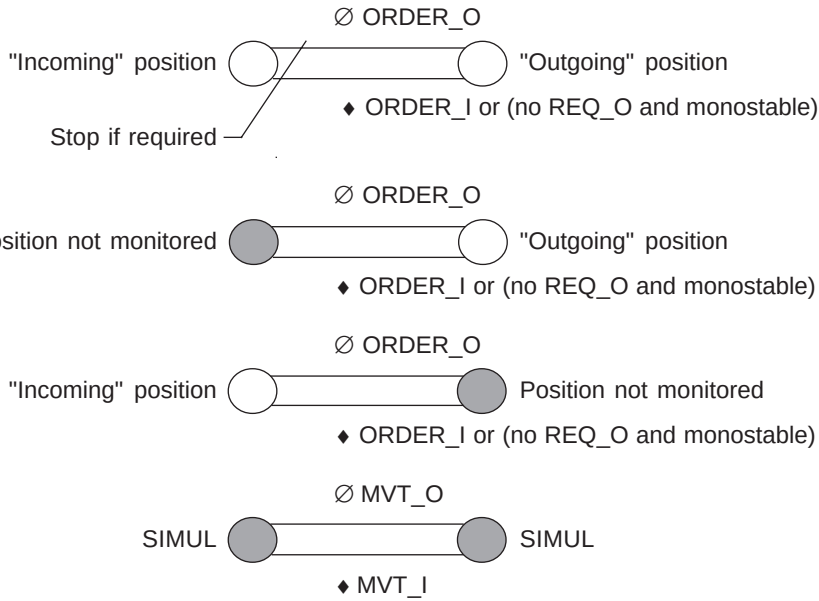
4.7-2 Executing the Movement

When operating normally (control/monitoring mode and READY bit = 1), the DFB controls the movement(s) by performing the following operations :

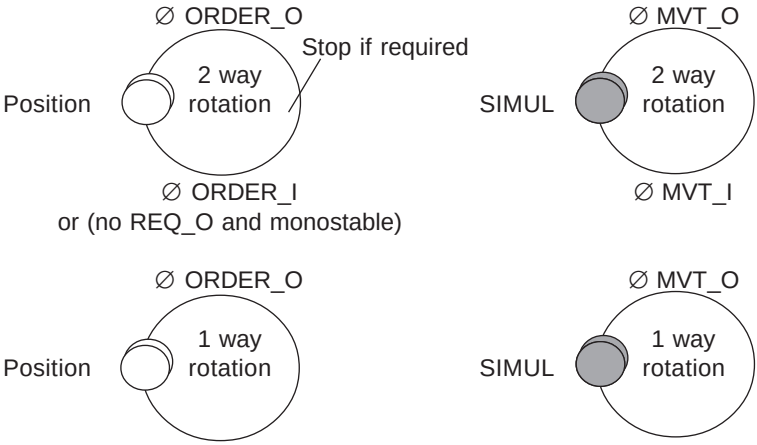
- monitoring sensor data (SENSOR_I and SENSOR_O inputs and NOSENS_I and NOSENS_O inputs if used)
- monitoring requests (REQ_I and REQ_O inputs)
- monitoring movement times
- recording maximum and minimum movement times
- learning movement times
- detecting and reacting to errors
- generating confirmation reports for functional control
- generating control orders for the actuator (ORDER_I and ORDER_O outputs)
- updating operation indicators
- assisting with restarting the cycle

The following diagrams indicate the different movement actions :

- **Translational**



- **Rotating**



Movement authorization
When movement requests are missing and if these have been authorized (the "request refused" information in STATUS0 has not been activated), the DFB sets the POSSIB_I and POSSIB_O outputs to 1.

Notes

- SECUPERM (permanent operating conditions) or SECU_O / I (safety conditions of the movement) enter into the evaluation of the POSSIB_O / I bit if their absence deactivates the orders, **ie. if the associated errors are selected in the RST_ORD mask**.
- A movement will be refused if an error selected in RST_ORD is present at the time of the request.
- When a movement is requested, the presence of the opposite request will always prevent its execution (this error cannot be masked). Furthermore, when a movement is being executed, an opposite request will cancel the order whether the request is accepted or not.
- For position control, a request has no effect on order type commands until the position is reached (type 3 or 4) : POSSIB takes this condition into account.

Disappearance of a sensor and starting the movement
For position control, the disappearance of a sensor is only indicated after the time indicated by DIS_TIME. This check is disabled as soon as a movement request is accepted.

Sensor not expected
Except for recalibration mode, the appearance of an unexpected sensor is only indicated after the time indicated by APP_TIME.

Information about the movement
The DFB sets data variables which provide information about the execution of the movement :

- The STATE_I and STATE_O outputs indicate the state of the movement monitored by the DFB (position reached). FAULT_I and FAULT_O indicate an error with the movement in progress.
- INC_I and INC_O indicate inconsistency between the position expected (RESEQ_I, RESEQ_O and ORIGIN data variables) and the STATE_I and STATE_O outputs, when there is no order or request.
- The MVT_I and MVT_O internal data variables indicate that the movement in progress has not yet ended (out of position).

During the movement, the safety conditions linked to the movement and the permanent conditions must remain valid according to the RST_FB and RST_ORD masks.

4.7-3 Recalibration Mode

Following an error configured in RST_FB or a RESET_FB request, which sets the DFB to recalibration mode, the DFB :

- deactivates the READY bit
- deactivates the STATE_I / O and ORDER_I / O outputs
- accepts the configuration data and continues operation if there are no configuration errors in STATUS1 (only in the case of RESET_FB requests)
- waits for an INIT request to clear the errors which are no longer present in STATUS0 (only in the case of an error). The DFB is then in a RESET state in which it is "frozen" : it no longer tests the permanent conditions and safety conditions, and its outputs stay the same.
- changes to recalibration mode to find a reference point again
- returns to control/monitoring mode as soon as it detects a consistent sensor configuration

4.7-4 Help with Restarting the Cycle

The RESEQ_I, RESEQ_O and ORIGIN data variables inform the DFB of the state which is expected by the control system. The DFB records the last expected state (RESEQ_I, RESEQ_O or ORIGIN set to 1). If the state or the movement monitored by the DFB is not consistent with the expected state (the last one to be recorded), the INC_I and INC_O outputs indicate an inconsistency. When the DFB changes to recalibration mode, the expected states before the change are recorded.

4.7-5 Recording the Maximum and Minimum Movement Times

For each movement executed, the DFB records (in non-simulated mode) the time and the maximum and minimum times in the TMIN_I, TMAX_I, TMIN_O and TMAX_O data variables. The maximum times are only recorded if they are less than the RMAX_I and RMAX_O maximum reference values.
The RESET_CT data variable enables the maximum and minimum values of the movements to be reset.

4.7-6 Learning the Movement Times

The DFB can learn the movement times. For this, the configuration data for time management must be initialized to 0.
When a movement is executed without interruption, the RMIN_O (or RMIN_I) data variable takes a value equal to **half the movement time**, whereas RMAX_O (or RMAX_I) takes a value equal to **1 1/2 times this value**. A movement is said to be executed without interruption when it is not stopped voluntarily either by lack of requests for the actuators enabling it, or by an error which resets the orders to zero.
When the times of the two movements have been acquired, the ADJ_TIME bit is set to 1.

4.7-7 Characteristics of Rotating Movement

Evaluation of the position
If the SENSOR_I and SENSOR_O (and NOSENS_I and NOSENS_O if used) inputs are not identical, a "command error" will be indicated.

For position control, if at least one of the two inputs changes to 0, the DFB will begin to measure the disappearance time of the sensor(s), until both inputs are reset to 1 at the same time.

For motion control, the position will be considered as "left" if both sensors are seen at 0 at least once at the same time. The position will be considered as "reached" if both sensors are seen at 1.

The only errors to be indicated concerning the sensors are as follows :

- for position control : "sensor(s) disappeared" or "sensor(s) not deactivated"
- for motion control : "sensor(s) untimely" or "sensor(s) delayed"

Request maintained and position reached
When rotating, only one position is monitored (both sensors are acting together). When at position and contrary to translational movement, both requests are accepted and start the two possible movements.

When a movement ends (position reached), and if the "incoming" or "outgoing" request is still present, the movement is automatically restarted. In order to prevent this with the rotating movement, the requests are interpreted on a rising edge.

4.7-8 Manual Mode

Execution of the movements in manual mode (independently of the machine cycle) is carried out by the operation command, independently of the DFB. This reacts to the commands in the same way as in automatic mode.

However, so that the DFB can operate in manual mode, it must also be executed independently of the machine cycle. Therefore, if manual control of the DFB is planned, it must be executed in a PL7 module which is easily accessible regardless of the status of the machine cycle : module executed on each PLC scan (POST or SR) which can be called easily, according to or independent of the machine cycle.

4.7-9 PLC Operating Modes

On an application transfer or change of cartridge, the DFB reinitializes all the data, takes the configuration data into account and returns to recalibration mode (READY at 0).

On a %S0 request or restart after a power outage, the DFB returns to recalibration mode (READY at 0). The ORDER_I / O and STATE_I / O outputs are reset to 0. The counters managed by RESET_CT and the reference times are preserved.

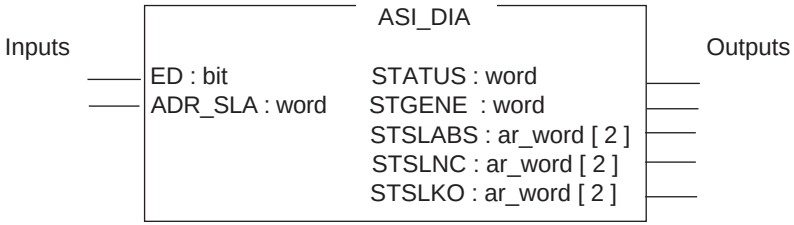
The control/monitoring mode is activated when a position is found, no error is indicated and no request is present (whatever the value of NEW_REQ).

5.1 General

This DFB is used to monitor the appearance of errors on the ASI bus :

- module or bus error
- slave(s) absent
- slave(s) not configured
- slave(s) failed

5.2 Presentation of ASI_DIA



ar_word [2] : table with two 16-bit words

Note
ASI errors are recorded in area 0.

5.3 Description of the Parameters

Input parameters

Parameter	Type	Access	Description
ED	bit	R (1)	DFB activation bit If ED = 0, the ASI bus is not monitored. The default value of ED is 0.
ADR_SLA	word	R (1)	Address XY of the ASI module Rack : X Module : Y

(1) : Access by program

Output parameters

Parameter	Type	Access	Description
STATUS	word	R (1)	Indicates the **type of error** detected : Bit 0 = 1: module or bus error Bit 1 = 1: slave(s) absent Bit 2 = 1: slave(s) not configured Bit 3 = 1: slave(s) failed The default value of STATUS is 0.
STGENE	word	R (1)	Specifies the **module or bus error** : Bit 0 = 1: ASI module not responding OK to the module identification request Bit 1 = 1: slave with address 0 detected on the bus Bit 2 = 1: ASI power supply fault Bit 3 = 1: OFFLINE phase active Bit 4 = 1: DATA_EXCHANGE mode not active Bit 5 = 1: no slave present on the bus The default value of STGENE is 0.

(1) : Access by program

Output parameters (continued)

Parameter	Type	Access	Description
STSLABS [ ]	ar_word [2]	R (1)	**List of absent slaves** **STSLABS[0] : slaves 0 to 15** Bit 0 : not significant, always at 0 Bit 1 = 1: the slave configured at address 1 is absent Bit 2 = 1: the slave configured at address 2 is absent … Bit 15 = 1: the slave configured at address 15 is absent **STSLABS[1] : slaves 16 to 31** Bit 0 = 1: the slave configured at address 16 is absent Bit 1 = 1: the slave configured at address 17 is absent Bit 2 = 1: the slave configured at address 18 is absent … Bit 15 = 1: the slave configured at address 31 is absent The default values of STSLABS[0] and STSLABS[1] are 0.
STSLNC [ ]	ar_word [2]	R (1)	**List of slaves not configured** **STSLNC[0] : slaves 0 to 15** Bit 0 : not significant, always at 0 Bit 1 = 1: the slave detected at address 1 is not configured Bit 2 = 1: the slave detected at address 2 is not configured … Bit 15 = 1: the slave detected at address 15 is not configured **STSLNC[1] : slaves 16 to 31** Bit 0 = 1: the slave detected at address 16 is not configured Bit 1 = 1: the slave detected at address 17 is not configured Bit 2 = 1: the slave detected at address 18 is not configured … Bit 15 = 1: the slave detected at address 31 is not configured The default values of STSLNC[0] and STSLNC[1] are 0.

(1) : Access by program

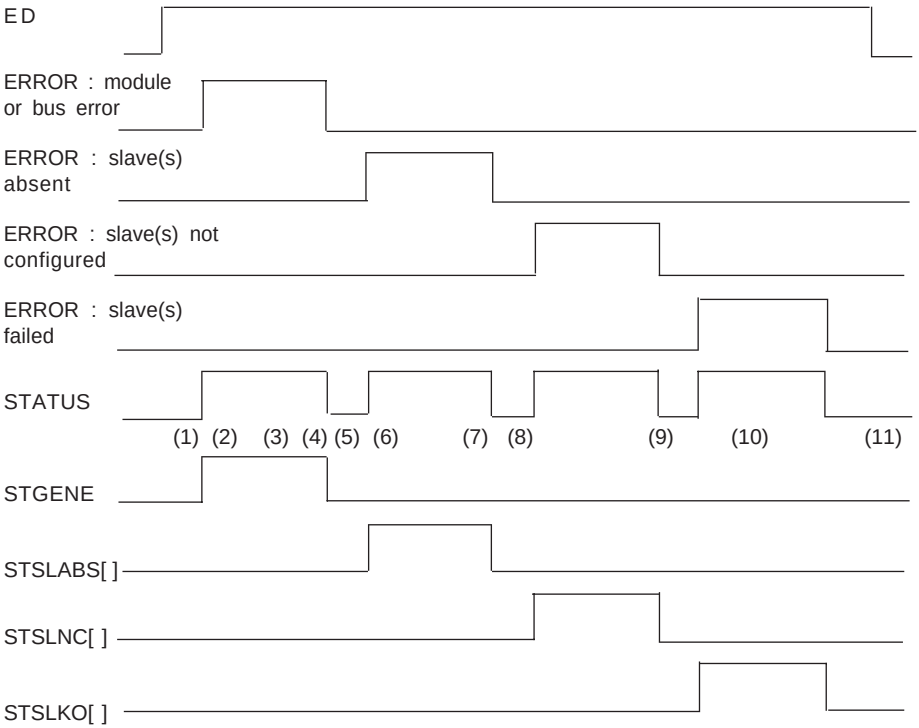
Output parameters (continued)

Parameter	Type	Access	Description
STSLNC [ ]	ar_word [2]	R (1)	**List of failed slaves** **STSLKO[0] : slaves 0 to 15** Bit 0 : not significant, always at 0 Bit 1 = 1: the slave at address 1 is incorrectly configured or failed Bit 2 = 1: the slave at address 2 is incorrectly configured or failed … Bit 15 = 1: the slave at address 15 is incorrectly configured or failed **STSLKO[1] : slaves 16 to 31** Bit 0 = 1: the slave at address 16 is incorrectly configured or failed Bit 1 = 1: the slave at address 17 is incorrectly configured or failed Bit 2 = 1: the slave at address 18 is incorrectly configured or failed … Bit 15 = 1: the slave at address 31 is incorrectly configured or failed The default values of STSLKO[0] and STSLKO[1] are 0.

(1) : Access by program

5.4 Operation

All the data used in the ASI_DIA DFB is obtained from the language objects associated with the ASI module. These language objects are read every second so that execution of the application is not slowed down.



(1) A **module or bus error** is recorded by the DFB if there is an break in the ASI power supply. Bit 0 of STATUS and bit 2 of STGENE are set to 1.

(2) A slave with address 0 is detected on the bus. The STGENE bit is set to 1.

(3) The ASI power supply has been restored but the “Module or bus error” is not cleared because a slave with address 0 is still detected on the bus.

(4) The slave with address 0 is no longer detected on the bus and the error has disappeared. The STATUS and lSTGENE words are set to 0.

5) The "Slave(s) absent" error is set in the STATUS word (bit = 1) and bit 10 of STSLABS[0] is set to show that the ASI slave with address 10 is absent.

(6) The ASI slave with address 14 is disconnected. Only bit 14 of STSLABS[0] is set to 1.

(7) The ASI slaves with addresses 10 and 14 are present again on the ASI bus. Bit 1 of STATUS is set to 0 and STSLABS[0] is at 0.

(8) The “Slave(s) not configured” error is set in the STATUS word (bit 2 = 1) and bit 12 of STSLNC[1] is set to show that an ASI slave with address 27 is present on the bus but not configured.

(9) Disappearance of the unconfigured slave with address 27. STATUS and STSLNC are at 0.

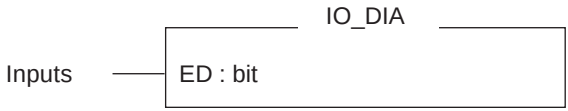
(10) The “Slave(s) not configured” error is set in the STATUS word (bit 3 = 1) and bit 5 of STSLKO[0] is set to show that an ASI slave with address 5 has failed.

(11) Disappearance of the error in the failed slave with address 5. STATUS and STSLKO are at 0.

6.1 General

This DFB is used to monitor the state of the I/O (based on the value of bit %S10). The message **I/O error** is displayed by the viewer. Acknowledgment is not required by the operator.

6.2 Presentation of IO_DIA



6.3 Description of the Parameters

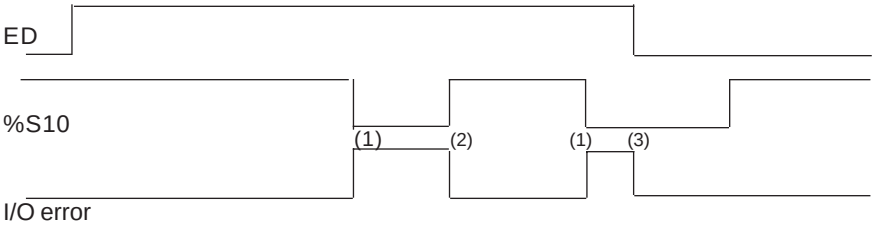
Input parameters

Parameter	Type	Access	Description
ED	bit	R (1)	DFB activation bit If ED = 1, bit %S10 (I/O error) is monitored. The default value of ED is 0.

(1) : Access by program

Note
I/O errors are registered in area 0.

6.4 Operation

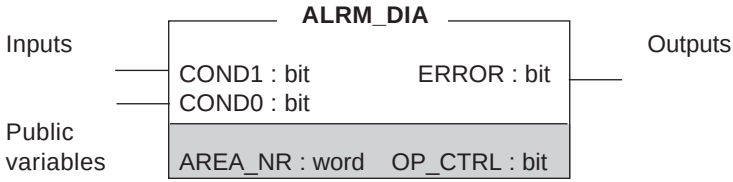


(1) An I/O error is detected when system bit %S10 is set to 0.
(2) The error is reset to zero when system bit %S10 is set to 1.
(3) The error is reset to zero when the ED input changes to 0.

7.1 General

This DFB is used to record errors in a diagnostic buffer. If input Cond1 changes to 0 or input Cond0 to 1, an error will be registered in the diagnostic buffer. If both Cond1 and Cond0 are incorrect, only one error is registered.
The error disappears when both inputs Cond1 and Cond0 return to their correct values.

7.2 Presentation of ALRM_DIA



7.3 Description of the Parameters

Input parameters

Parameter	Type	Access	Description
COND1	bit	R (1)	Input bit to be monitored for state 1. If the DFB is executed and if this bit changes to 0, the DFB will show an error. If the COND0 input changes to 1, no new errors are recorded. The default value of COND1 is 1.
COND0	bit	R (1)	Input bit to be monitored for state 0. If the DFB is executed and if this bit changes to 1, the DFB will show an error. If the COND1 input changes to 0, no new errors are recorded. The default value of COND0 is 0.

(1) : Access by program

Output parameters

Parameter	Type	Access	Description
ERROR	bit	R (1)	Error bit. This bit is set to 1 when an error occurs. This bit is set to 0 when the ED input changes to 0 or if no errors occur.

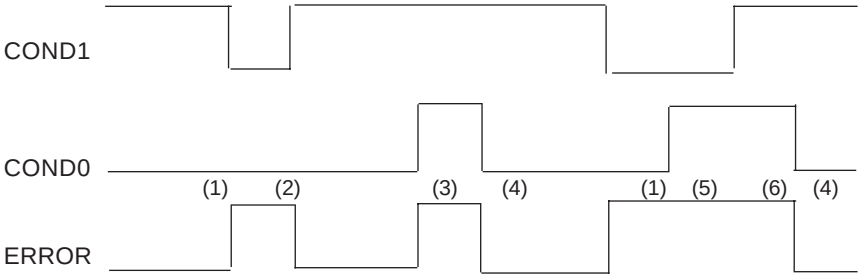
(1) : Access by program

Public variables

Parameter	Type	Access	Description
AREA_NR	word	R (1)	This word is used to specify which area of the control system is monitored by the diagnostic DFB. Example : Machining : no.1 Milling : no. 2 Tapping : no. 3 AREA_ NR must have the value 1, 2 or 3 so that the user can identify the faulty part of the control system. The above areas should be divided in the same way as the function modules. AREA_ NR can take a value from 0 to 15 (the default value is 0).
OP_CTRL	bit	R (1)	This bit indicates whether or not acknowledgment of the DFB instance by the operator is required. OP_CTRL = 0 : no acknowledgment by the operator OP_CTRL = 1 : acknowledgment by the operator The default value of OP_CTRL is 0.

(1) : Access by program

7.4 Operation



(1) An error is detected when the COND1 input is set to 0.

(2) The error is reset to zero when the COND1 input is set to 1.

(3) An error is detected when the COND0 input is set to 1.

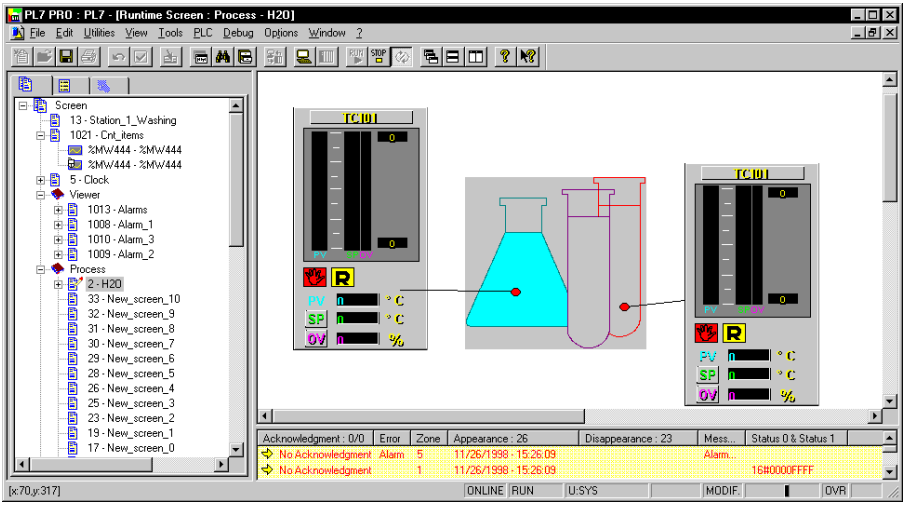
(4) The error is reset when the COND0 input is set to 0.

(5) An error is not detected when the COND0 input is set to 1 because an error is already present.

(6) The error is not reset when the COND1 input is set to 1 because the COND0 input is still at 1.

8.1 Presentation

The Viewer is used to debug an application and view any faults in the application .



Display window

All error messages appear in a window in the bottom right of the screen browser (below the graphics editor). It is possible to modify the size of this window (using the mouse only) but its location cannot be changed. The window can also be hidden.
The display window contains a list of messages and may have two scroll bars : a vertical scroll bar if there are more messages in the list than can be displayed, and a horizontal scroll bar if the Viewer is not wide enough to display the entire contents of a line.

8.2 Composition of Messages

Each line displayed in the viewer corresponds to a fault and contains the following information :

- **icon plus text which shows the status of the message** (the message needs to be acknowledged, has been acknowledged or does not require acknowledgment)
- **the type of faulty DFB**
- **the geographical area in which the fault originated**
- **the date and time at which the fault appeared**
- **the date and time at which the fault disappeared**
- **the message associated with the fault**
- **the value of the status word at the time of the fault**

The list is thus split into seven columns whose size can be modified (increased or reduced) by the user (with the mouse). If a column is not wide enough to display all the information it contains, the line ends in 3 dots. The width of each column is saved and restored when the Runtime Screens tool is opened. The column headings also provide information on the number of messages and their status.

8.3 Displaying Messages

The list of messages can be sorted according to each of the fields contained in the list.

To sort the list, simply click on the title of the column containing the data to be sorted. A second click will sort the list in reverse order (method similar to that of Windows Explorer).
By default, messages are inserted into the list in the chronological order in which the faults appeared.

Warning : Even if the list is sorted according to a given field, new messages always appear at the end of the list.

The number of messages which can be displayed in the list is limited only by the memory available. If there is insufficient memory, a message warns the user, and messages related to faults which have disappeared AND been acknowledged (if they have to be) are then deleted.
The color of the messages (text and background color) and the flashing associated with a message requiring acknowledgment can be modified in order to clearly distinguish the different types of messages displayed in the Viewer window.

When a message appears, the split bar flashes until the message is displayed.

Colors :

The information displayed in the Viewer uses the default colors of Windows. They are defined in the Display tab, Display Properties dialog box, Windows elements (text + background colors) and selected elements (text + background color), and are similar to those used in Windows Explorer.

By default, the text of a fault which has appeared (and not yet disappeared) is in red. It is, however, possible to choose another color (for both the text and background) using the Configuration box in the Viewer tab (see section 2.5).

When the fault disappears, the line containing the message returns to the standard colors.

Flashing :

A message requiring acknowledgment can be made to flash in two different ways : basic flashing (only the icon showing the acknowledgment status flashes) and extended flashing (the whole line flashes).

If the whole line flashes, background and text colors are simply reversed during flashing.

In the Viewer, it is possible to display only those messages relating to a specific zone or zones. This can be specified in the Configuration box of the Viewer tab. The number of zones is between 0 and 15. By default, all messages (regardless of their zone) are displayed in the Viewer.

8.4 Operation and Management of Messages

Browsing :
It is possible to browse through the list of messages using the UP, DOWN, PAGEUP, PAGEDOWN, HOME and END keys on the keyboard or using the mouse (use the scroll bar if the list contains more messages than can be displayed).

Acknowledgment :
To acknowledge a message which requires it, simply select the message and use the appropriate item from the shortcut menu by clicking with the right mouse button. It is also possible to use the F10 function key or the button on the utilities toolbar. Several messages can be acknowledged simultaneously (multiple selection). When a message is acknowledged, a command is sent to the PLC and the associated "box" icon is checked. A message can be acknowledged by another Viewer. In this case, the Runtime Screens tool is warned of this and the message is displayed as acknowledged.

Deleting messages from the list :
A message requiring acknowledgment or a message which has not disappeared cannot be deleted.
The Delete key or the appropriate shortcut menu item can be used to delete only those messages which have disappeared AND been acknowledged (if required).

Properties (status) :
It is possible to display a dialog box containing more precise information on the alarm message and its status by pressing the ENTER key or using the shortcut menu.
The following information is displayed :
- The name of the instance and the type of faulty DFB
- The address of the program containing the instance of the faulty DFB : Task + Section (if Appli DFB)
- The associated text and status bits

Activating another MDI tool :
If one (or more) messages are selected, the following MDI tools can be activated from the PL7 software workshop :

- Animation tables to display the external and internal data of the DFB instance (F6 function key)
- Cross references (F7 function key)
- The language editor where the instance of the faulty DFB is referenced, or the configuration editor if it is a system DFB (F8 key)

These tools are activated using the shortcut menu (click with the right mouse button), the function keys (F6, F7 and F8) or the buttons on the Utilities toolbar.

Behavior of the Viewer :

- When the Runtime Screens tool is activated, the display window is initialized (messages are not saved from one session to the next). If messages are already present in the diagnostic buffer when the connection is made, these are inserted in the list.
- When connecting to the PLC (switching to online mode), transferring a program to the PLC or during a reconfiguration operation, the messages present in the list are deleted. However, if messages are already present in the diagnostic buffer when the connection is made, these are inserted in the list.
- On disconnection, messages remain displayed in the Viewer. However, messages requiring acknowledgment no longer flash and they can no longer be acknowledged.

These functions can be accessed from the Viewer shortcut menu, which is displayed in the Viewer window, by clicking with the right mouse button.

8.5 Archiving Messages

Messages are archived, thus creating a log file. The archiving function and file location are configured in the Configuration dialog box, Viewer tab.

The directory in which the log file is located can be changed. This file is called **AppName.his** (AppName being the name of the current application) and by default it is located in the PL7 source directory (SRC).

Archiving procedure :
Messages are archived as lines (as soon as a message is read in the PLC buffer, it is written to the file). If a message appears then disappears, it is represented by a single line (message) in the display window, but by two lines in the log file.
To prevent the file from becoming too large, it is renamed as AppName.BAK after every 1000 records, and a new log file is recreated with its original name.

Warning : If a .BAK file already exists, it is deleted without warning.

This file is in ASCII format (each data item is separated by a ;). It can therefore be easily imported into any word processing or spreadsheet package.

9.1 Displaying active alarms

Informations relative to an alarm are as follows :

- **its sequence number**
- **the date and time when the error condition occurred and ended,**
- **the type of alarm** : EV_DIA, MV_DIA, NEPO_DIA or ASI_DIA ,
- **the alarm statuses**,
  - -ACK : alarm acknowledged (seen by the operator), the alarm number is fed back to the PLC,
  - -ON : alarm not acknowledged.
- **the message assigned to the alarm**.

```
 * ACTIVE ALARMS : 011 *
002 03/04/97 11:07:54....NEPO....ACK
Failure left  motor N°3
<ALT>+<P> ->  Status <↵>  -> return
```

This screen also displays the number of activated alarms and the sequences of keys which can be used to browse in the CCX17 screens.

From the application screen, pressing
[ALT] + [ACK] displays the list of active alarms, moving to the most recent.

[↑] [↓] Moves the cursor in the list. The selected alarm is displayed in reverse video.

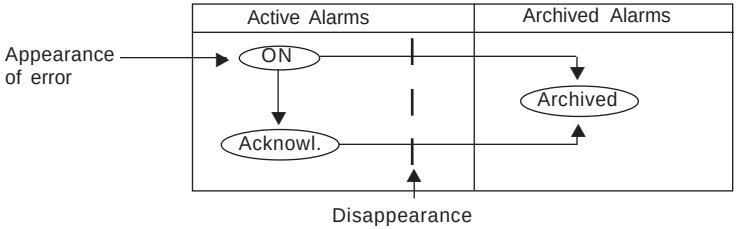
[ALT] + [↓] moves the cursor to the end of the list.

[ALT] + [↑] moves the cursor to the start of the list.

[ACK] allows the user to acknowledge the alarm selected. For a DFB type alarm, the acknowledgement information is sent to the PLC, if this option has been configured in the DFB instance.

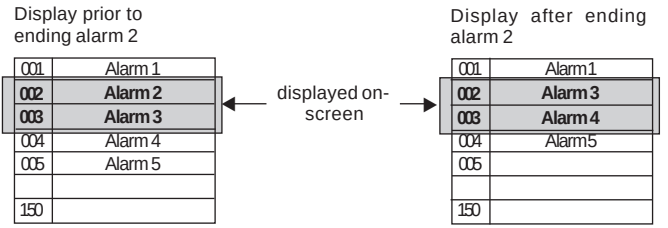
[ALT] + [P] if the alarm comes from a diagnostic OFB or DFB, the status screen is displayed.

[↵] used to exit consultation mode and return to the current dialog screen.



Remark on displaying active alarms

The alarms are saved in the memory field as they occur with an occurrence number. The screen provides a real-time display of a group of alarms and when a displayed alarm above the displayed field is removed from the list, it is compacted (see the example below).



- **Displaying the OFB or DFB status**

Information on the status of the OFB which has generated the alarm, is displayed on two lines. To display all the messages, use the **[↑] [↓]** keys. If the last message is displayed (last status bit), only the **[↑]** key appears. Similarly if the first message is displayed, only the **[↓]**key appears.

As well as the status messages, the screen displays :

- The symbol (SDBASE) of the error bit which generated the error (OFB only),
- The name of the DFB instance (DFB only),
- The message associated with the alarm.

  If the fault has disappeared (OFB error bit = 0), the message "ALARM DISAPPEARED" is displayed and the status messages are deleted.

10.1 General

PL7 software (version ≥ 3.4) allows users to create their own diagnostic DFBs.

Using 2 model DFBs, users can create up to :
- 26 types of process type user diagnostic DFBs,
- and 26 types of system type user diagnostic DFBs.

These DFBs have a predefined name :
- Usra_dia, Usrb_dia, …, Usry_dia and Usrz_dia for process DFBs.
- Sysa_dia, Sysb_dia, …, Sysy_dia and Sysz_dia for system DFBs.

Process DFB : the code created by the user will be oriented towards process control and application monitoring (example : monitoring levels with several thresholds).

System DFB : the code created by the user will be oriented towards module and system monitoring (example : monitoring an axis control module, monitoring system bits and words etc).

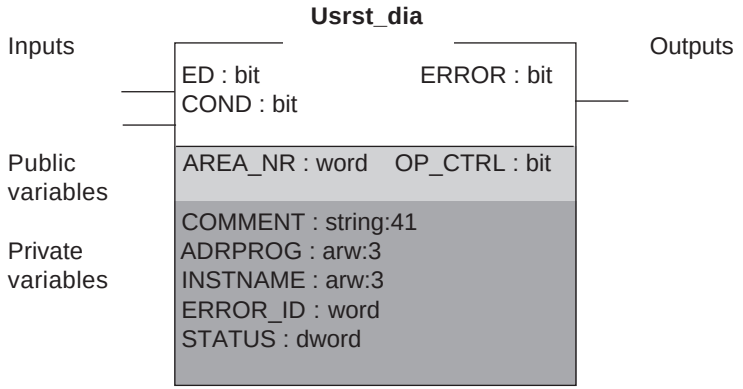
These DFBs can be easily integrated into the diagnostic offer : they are seen by the CCX17 Viewer (within the limits of the memory available) and by the Runtime Screens Viewer.

Depending on the error message selected, using the command “Open the associated editor F8 ”, the Runtime Screens Viewer displays :
- The rung or statement which calls the faulty instance for process DFBs.
- The hardware configuration screen for system DFBs.

10.2 Description of model DFBs

In order to create user diagnostic DFBs, PL7 software offers 2 model DFBs :
- One written in Ladder language, called Usrld_dia.ufb.
- The other written in Structured Text language, called Usrst_dia.ufb

These DFBs have comments attached (descriptive file), and are protected (password : diaguser) and supplied in binary format.
Using one of these models, the user can enhance the interface and code in order to create the diagnostic DFB most suited to his application.



Input parameters of the model DFB

Parameter	Type	Access	Description
ED	bit	R (1)	Activation bit of the DFB. If ED = 0, the DFB is not executed. The default value of ED is 0.
COND	bit	R (1)	Input bit to be monitored at state 1. If the DFB is executed and if this bit changes to 0, the DFB displays an error. The default value of COND is 1.

(1) Access via the program

Output parameters of the model DFB

Parameter	Type	Access	Description
ERROR	bit	R (1)	Error bit. This bit is set to 1 when an error appears This bit is set to 0 when the input ED changes to 0 or if there are no errors.

(1) Access via the program

Public variables of the model DFB

Parameter	Type	Access	Description
AREA_NR	word	R (1)	This word is used to specify which area of the control system is monitored by the diagnostic DFB. AREA_ NR can have a value of between 0 and 15 (the default value is 0).
OP_CTRL	bit	R (1)	This bit indicates whether or not acknowledgment of the DFB instance by the operator is necessary. OP_CTRL = 0 : no acknowledgment by the operator OP_CTRL = 1 : acknowledgment by the operator The default value of OP_CTRL is 0.

(1) Access via the program

Private variables of the model DFB

Parameter	Type	Access	Description
COMMENT (2)	string:41	R (1)	Error message displayed by the viewer : contains the instance comment entered in the variable editor.
ADRPROG (2)	arw:3	R (1)	Address of the program section calling the faulty DFB instance. This address is used by the viewer to display the rung or statement.
INSTNAME (2)	arw:3	R (1)	Name of the faulty instance. Used to display the faulty instance.
ERROR_ID	word	R (1)	Identification no. of the error sent back by the diagnostic buffer (used to deregister an error).
STATUS	dword	R (1)	Report of the error to be managed by the user.

(1) Access via the program
(2) Compulsory variables : do not modify

10.3 Example DFBs

Examples of diagnostic DFBs are provided :
- 1 example of a process DFB implementing a bar chart (Usra_dia.dfb)
- 2 examples of system DFBs for diagnosis of an axis module (Sysa_dia.dfb) and Fipio diagnosis (Sysb_dia.dfb)

These DFBs have comments attached (descriptive file), and are protected (password : diaguser) and supplied in binary format. They can, of course, be run and modified by the user.

10.4 Procedure for setting up a diagnostic DFB

The table below describes the procedure for creating a user diagnostic DFB.

Step	Action
1	Create a default application (by declaring a version ≥ V3.3 processor) and configure the diagnostic option (see section 1.3-1).
2	Import the binary file of a model DFB (Usrld_dia.ufb or Usrst_dia.ufb) : • Using the shortcut menu (right click on the DFB Types directory in the browser), select the **Import Binary** command. • Select the model file in the **DIAG subdirectory** which is located in the PL7 installation directory (example C:\PL7\PL7PRO33\DIAG), and click on **Import**.
3	Unprotect the DFB : • Double click on the model DFB imported into the browser. • Access the **Properties** command by right clicking in the editor window. • Select the **Unprotected** button, enter the password (diaguser) and confirm. • Confirm the DFB.
4	Rename the model DFB in the browser : Select the model DFB, click with the left mouse button on the name and rename this DFB : • Usr ?_dia for a process DFB • Sys ?_dia for a system DFB ? being a letter of the alphabet (for example : Usrf_dia, Sysb_dia)
5	Modify the interface of the DFB according to the DFB required. **Rules :** • The private variables "Comment", "Instname" and "Adrprog" are **compulsory** and neither their name nor type can be changed. • There must also be a DWORD type status variable (private or public). This variable may or may not be managed by the DFB (but it must exist) and can be given any name. • There must also be at least one WORD type error variable (private or public). This variable must be managed by the DFB (registration and deregistration) and can be given any name. • The initial value of the "Comment" variable is the default error message generated by this type of DFB. This message can be modified. **Warning :** Failure to observe the above rules may result in serious malfunctioning of all diagnostic DFBs.
6	Modify the code of the model DFB according to the DFB required and modify the value of the error class depending on the name of the DFB. The error class is specified in the parameter in the code of the REGDFB instruction : • 16#004A (Usra_dia) to 16#0063 (Usrz_dia) for process DFBs. • 16#008A (Sysa_dia) to 16#00A3 (Sysz_dia) for system DFBs. See list of REGDFB and DEREG error classes and instructions in section 10.5.

Step	Action
7	Confirm the DFB.
8	Test and debug the DFB code. **Note :** A diagnostic DFB requires a label in the language element (rung or statement) which contains its call.
9	Update the descriptive file (optional).
10	Protect (optional) the DFB (read or modify) : any password can be used. **Warning** The DFB must be protected before it is exported : if this is not done, the DFB irreversibly loses its diagnostic attributes.
11	Export the DFB in **binary** format : • Using the shortcut menu (right click on the DFB in the browser), select the **Export Binary** command. • Select the **DIAG subdirectory** which is located in the PL7 installation directory (example C:\PL7\PL7PRO33\DIAG) • Name the DFB : name_of_DFB_type.ufb (eg : Usra_dia.ufb) and save. The DFB can then be reinserted into any user application using a binary import.

Warning

Failure to observe the above rules or incorrect coding of the DFB may result in serious malfunctioning of all diagnostic functions.

10.5 Recording alarms in the diagnostic buffer

10.5-1 Recording an alarm

The alarm registration instruction REGDFB, entered in the DFB code, stores and time stamps an alarm in the diagnostic buffer.

Syntax :

REGDFB (Area_nr, Class, Slen, Op_ctrl, Comment, Instname, Adrprog, Status, Error_id, Stat)

Input parameters :

Name	Role	Type
Area_nr	Area of the machine monitored by the DFB : 0 to 15	WORD
Class	Error class : (see table on next page) • 16#004A to 16#0063 for process DFBs • 16#008A to 16#00A3 for system DFBs	WORD
Slen	Status length : 0, 2 or 4 bytes 0 = no status managed 2 = status managed on a word 4 = staus managed on a double word	WORD
Op_ctrl	1= Operator acknowledgment required 0= no acknowledgment	BOOL
Comment	Default error message associated with the DFB instance	STRING
Instname	Name of the faulty instance	AR_W
Adrprog	Program address of the faulty DFB instance	AR_W
Status (1)	DFB status	DWORD

(1) Declared in the OUT parameter so that it is entered by address and not value. Must be updated by the DFB.

Output parameters :

Name	Role	Type
Error_id	Error identifier	WORD
Stat (1)	Error registration report	WORD

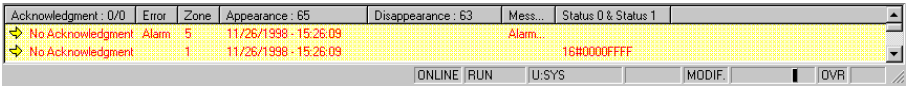
(1) System word %SW160 is reserved to receive the result of the registration of the diagnostic DFBs (its use is not obligatory but advisable).

Possible values of the Stat parameter :
- If registration is successful : Stat = 0 and Error_id is enabled
- If registration fails : Error_id is disabled
  Stat = 1 diagnostic buffer not configured
  Stat = 2 diagnostic buffer full

Error class (Class parameter) **:**
This parameter must be entered by the user. The error class code no. must correspond with the name given to the diagnostic DFB type. The table below shows the correspondence between the DFB type name and the Error class no.

Process DFB				**System DFB**			
Name	Code	Name	Code	Name	Code	Name	Code
Usra_dia	16#004A	Usrq_dia	16#005A	Sysa_dia	16#008A	Sysq_dia	16#009A
Usrb_dia	16#004B	Usrr_dia	16#005B	Sysb_dia	16#008B	Sysr_dia	16#009B
Usrc_dia	16#004C	Usrs_dia	16#005C	Sysc_dia	16#008C	Syss_dia	16#009C
Usrd_dia	16#004D	Usrt_dia	16#005D	Sysd_dia	16#008D	Syst_dia	16#009D
Usre_dia	16#004E	Usru_dia	16#005E	Syse_dia	16#008E	Sysu_dia	16#009E
Usrf_dia	16#004F	Usrv_dia	16#005F	Sysf_dia	16#008F	Sysv_dia	16#009F
Usrg_dia	16#0050	Usrw_dia	16#0060	Sysg_dia	16#0090	Sysw_dia	16#00A0
Usrh_dia	16#0051	Usrx_dia	16#0061	Sysh_dia	16#0091	Sysx_dia	16#00A1
Usri_dia	16#0052	Usry_dia	16#0062	Sysi_dia	16#0092	Sysy_dia	16#00A2
Usrj_dia	16#0053	Usrz_dia	16#0063	Sysj_dia	16#0093	Sysz_dia	16#00A3
Usrk_dia	16#0054			Sysk_dia	16#0094		
Usrl_dia	16#0055			Sysl_dia	16#0095		
Usrm_dia	16#0056			Sysm_dia	16#0096		
Usrn_dia	16#0057			Sysn_dia	16#0097		
Usro_dia	16#0058			Syso_dia	16#0098		
Usrp_dia	16#0059			Sysp_dia	16#0099		

Correspondence between input parameters and the various viewer areas :



Area name	**Correspondence**
Acknowledgment	Input parameter : Op_ctrl and acknowledgment order from the viewer
Fault	Name of the faulty DFB type (access the instance name Instname by right clicking on the faulty DFB and using the Property command)
Area	Input parameter : Area_nr
Appearance	Time-stamping performed when the REGDFB command is executed
Disappearance	Time-stamping performed when the DEREG command is executed
Message	Input parameter : Comment
Status 0 & 1	Input parameter : Status

10.5-2 Deregistering an alarm

The alarm deregistration instruction DEREG, entered in the DFB code, time-stamps the disappearance of the error in the diagnostic buffer.

Note : The alarm remains stored in the diagnostic buffer until the error is acknowledged (for errors requiring acknowledgment) and is read in all viewers.

Syntax :

Result:=**DEREG**(Error_id)

Input parameter :

Name	Role	Type
Error_id	Identifier of the error previously registered.	WORD

Return of the function :

Name	Role	Type
Result (1)	Error deregistration report : • If deregistration is successful, Result = 0 • If deregistration fails : - Result = 1 diagnostic buffer not configured - Result = 21 incorrect error identifier - Result = 22 no error registered with this identifier	WORD

(1) System word %SW161 is reserved to receive the result of the deregistration of the diagnostic DFBs (its use is not obligatory but advisable).