SIEMENS

SIMATIC S5

COM 552

Programming Package for the CP 552 Diagnostic Processor

Manual Volume 1/2

Order No. 6ES5998-3SE22
Release 05

Volume 2
User's Guides

Warnings
ESD Guidelines
Notes
Remarks Forms

C79000-H8576-C668-05

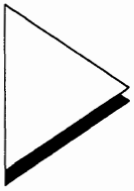
SIEMENS

SIMATIC S5

How to Use this Manual

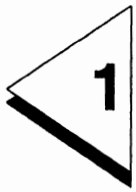
Notes

C79000-D8576-C668-05



Contents

Important Information



Contents

This manual provides support when using your **CP 552 diagnostic processor** with the **DIAGNOSIS software package.** Using the CP 552 and its software package, you can perform the following functions:

- **diagnose process errors**
- **display process control and system messages.**

CP 552-1 or CP 552-2

The CP 552 diagnostic processor is available in two versions. The CP 552-1 consists of **one** board. The CP 552-2 consists of **two boards** and has a **640 Kbyte buffered CMOS-RAM submodule**.

Note:

You **cannot** use the CP 552 with the **S5 -135U/S processor (CPU 921)** programmable controller.

Note:

In **multiprocessor operation:** you require **one CP 552 per CPU.**

Note:

Your can use your CP 552 and its software package **only** with the operating system S5-DOS **from version 3.x** onwards.

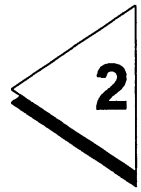
Note:

If you make changes to your STEP® 5 user program in **package 3** "Information and Special Functions" in the function "**rewiring**" your setpoint data elements are automatically updated.

Note:

For the diagnosis of process errors, you require the **standard function blocks for process error diagnosis with the CP 552 diagnostic processor**.

We wish you every success with your diagnostic processor!



Where to find what in the Manual?

Contents

This manual

COM 552 programming package for the CP 552 diagnostic processor

consists of two volumes. In each volume you will find an overview of the contents of the complete manual.

Volume 1

Part 1 **How to use this Manual**

these are general notes which you should read carefully.
This part also includes the following:

Abbreviations,
Ordering data,
List of further reading

Part 2 **Instructions**
CP 552 Diagnostic Processor

this part serves as an introduction. It contains important information about the **CP 552 module**, for example **installation** and **operation**.

Part 3

User's Guide
Process Error Diagnosis with the CP 552

this provides the following **basic information on process error diagnosis:**

- terms,
- functions in process error diagnosis
- error displays at the programmer
- single and group diagnosis
- installation
- displaying process control messages
- CP 552 start-up
- blocks on the CPU

Part 4

Practical Example
Process Error Diagnosis with the CP 552

this is an **introductory example**. You will use the example of an automatic carwash to create the setpoint data for process error diagnosis with the CP 552 and will start the process error diagnosis.

Part 5

empty

Volume 2

Part 1

User's Guide
Notes on the Operating Systems

before installing the DIAGNOSIS software package, you should read this guide carefully.

Part 2

User's Guide
Programming Package LAD, CSF, STL with Process Error Diagnosis

this guide describes how you generate the appropriate setpoint data in the LAD, CSF, STL package after you have input a segment.

Part 3

User's Guide
Error Displays on the Local Monitor

is an introduction to displays on the local monitor.

Part 4

User's Guide
COM 552 Programming Package

this is an introductory guide to the programming package COM 552. This part also describes the setpoint data editor, illustrates the **screen forms** and explains the fields and the permitted input.

Part 5 **User's Guide**
Displaying Process Control Messages

contains general information about planning process control messages either with or without the programming package COM PMC.

Abbreviations

Contents

A

AS 511	Serial PG interface

B

BSTACK	Block stack

C

COM 535	Programming package
COM 552	Programming package
COM PMC	software package PROCESS MONITORING AND CONTROL SYSTEM
CP 552	Diagnostic processor
CP 552-1	Diagnostic processor (consists of one board)
CP 552-2	Diagnostic processor (consists of two boards with a 640-Kbyte buffered CMOS-RAM submodule)
CP 535	Communications processor in the PLC for SINEC H1
CP 536	Communications processor in the PG for SINEC H1
CPU	Central processing unit

D

DB	Data block
DI	Digital input
DQ	Digital output

E

ENDP	Endpoint
EPROM	Erasable, programmable read-only memory
EU	Expansion unit

F

F	Flag
FB	Function block
FD	Drive (floppy disk or hard disk)

H

HDB (HTB)	Handling block

I

I	Input
IF	Device interface
IP	Intelligent peripheral (I/O) module
ISTACK	Interrupt stack

J

J...	Switch group or location on the CP 552 module

K

KOR C	923C coordinator

L

LAD, CSF, STL	STEP® 5 package
LED	Light-emitting diode

M

mot	Motive
MUX	Multiplexer

O

OB	Organization block

P

PB	Program block
PG	Programmer
PII	Process image of the inputs
PIQ	Process image of the outputs
PLC (or PC)	Programmable controller
PROT	Protected
pul	Pulse-dependent

Q

Q	Output

R

RAM	Random Access Memory
RO	Read only
RW	Read/write

S

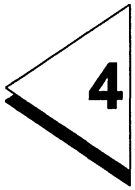
S5-Komi	S5-DOS command interpreter
SINEC H1	SIEMENS network communication - high range
SSNR	Interface number
SYSID	System identification area of the module

T

TTY interface	Serial current loop interface (20 mA)

X

X...	Jumper on the CP 552 module



Ordering Data

Contents

1. **CP 552-1** consisting of one PCB

   See catalog ST 54.1 for order number

2. **CP 552-2** consisting of two PCBs and a buffered CMOS-RAM submodule

   See catalog ST 54.1 for order number

3. **Software package DIAGNOSIS/ST**

   consisting of the following:

   programming package COM 552
   manual COM 552 / CP 552

   See catalog ST 59 for order number

4. **Manual COM 552 / CP 552**

   can be ordered separately

   See catalog ST 54.1 for order number

5. Cable connectors (not for S5-150U)

See catalog ST 54.1 for order number

Length key for 6ES5 cable connectors:

1.00 m	BB0
1.60 m	BB6
2.00 m	BC0
2.50 m	BC5
3.20 m	BD2
5.00 m	BF0
8.00 m	BJ0
10.00 m	CB0
12.00 m	CB2
16.00 m	CB6
20.00 m	CC0
25.00 m	CC5
32.00 m	CD2
40.00 m	CE0
50.00 m	CF0
63.00 m	CG3
80.00 m	CJ0
100.00 m	DB0
120.00 m	DB2
160.00 m	DB6
200.00 m	DC0

250.00 m	DC5
320.00 m	DD2
400.00 m	DE0
500.00 m	DF0
630.00 m	DG3
800.00 m	DJ0
1000.00 m	EB0

6. Cable connectors for S5- 150U

See catalog ST 54.1 for order number

7. Handling blocks

for S5- 135U / R processor

See catalog ST 57 for order number

for S5- 150U

See catalog ST 57 for order number

for S5- 155U

See catalog ST 57 for order number

8. Standard function blocks for error process diagnosis with the CP 552 diagnostic processor for

S5-115U
S5-135U
S5-150U
S5-155U

See catalog ST 57 for order number

9. Adapter casing in the S5-115U

See catalog ST 52.3 for order number

10. 923C coordinator for the S5-135U

See catalog ST 54.1 for order number

11. Programmer-multiplexer 757

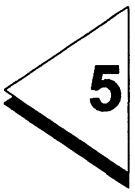
See catalog ST 59 for order number

12. Signalling functions for the standard CP

See catalog ST 57 for order number

13. Programming package COM PMC

See catalog ST 57 for order number



Further Reading

Contents

/1/ **CP 552 Standard Function Blocks**
Order no. C79000-G8576-C592

Descriptions of the standard function blocks for process error diagnosis with the CP 552 diagnostic processor for the following:

S5-115U
S5-135U
S5-150U
S5-155U

/2/ **PMCPRO**
Order no. C79000-G8576-C895

Description of the signalling functions for the standard CP
Description of the programming package COM PMC

/3/ **CP 535 Communications Processor**
Order no. 6ES5 998-0DG21

Manual for the communications processor CP 535
with the programming package COM 535

/4/ Operating instructions for the programmer-multiplexer 757

/5/ The manual supplied with your programmer

/6/ Description of the handling blocks relevant to your PLC

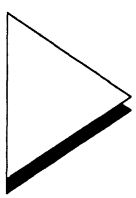
SIEMENS

SIMATIC S5

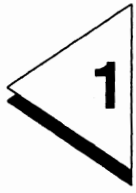
CP 552
Diagnostic Processor

Instructions

C79000-B8576-C669-05



Contents



Application and Mode of Operation

Contents

Using the **CP 552** diagnostic processor and the appropriate software package, you can **diagnose process errors** and **display process control and system messages**.

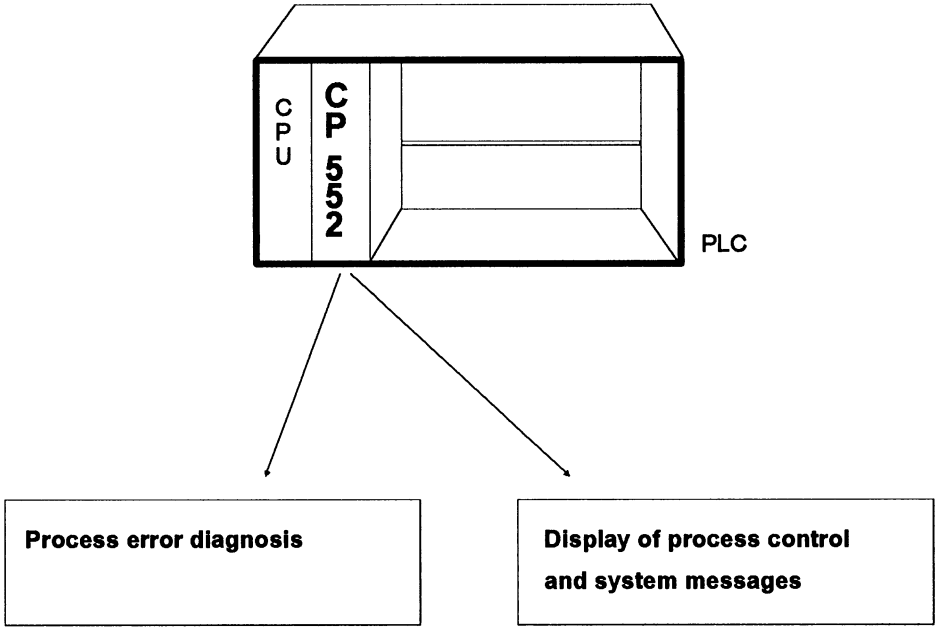


Fig. 1-1 Application

1.1 Diagnosing Process Errors

In automation engineering, the **diagnosis of process errors** is becoming increasingly important in the **reduction of installation time** and in the **reduction of downtimes**. Statistical studies clearly show that the most frequent downtimes in automated processes result from errors (faults) outside the programmable controller (PLC). The problems occurring in the programmable controller itself represent only a few percent of the total failures/faults.

Using the **CP 552 diagnostic processor** and the corresponding programming package, you can localize errors or faults occurring outside the programmable controller, for example, defective plant components, sensors, actuators, and wire breaks, much faster than before.

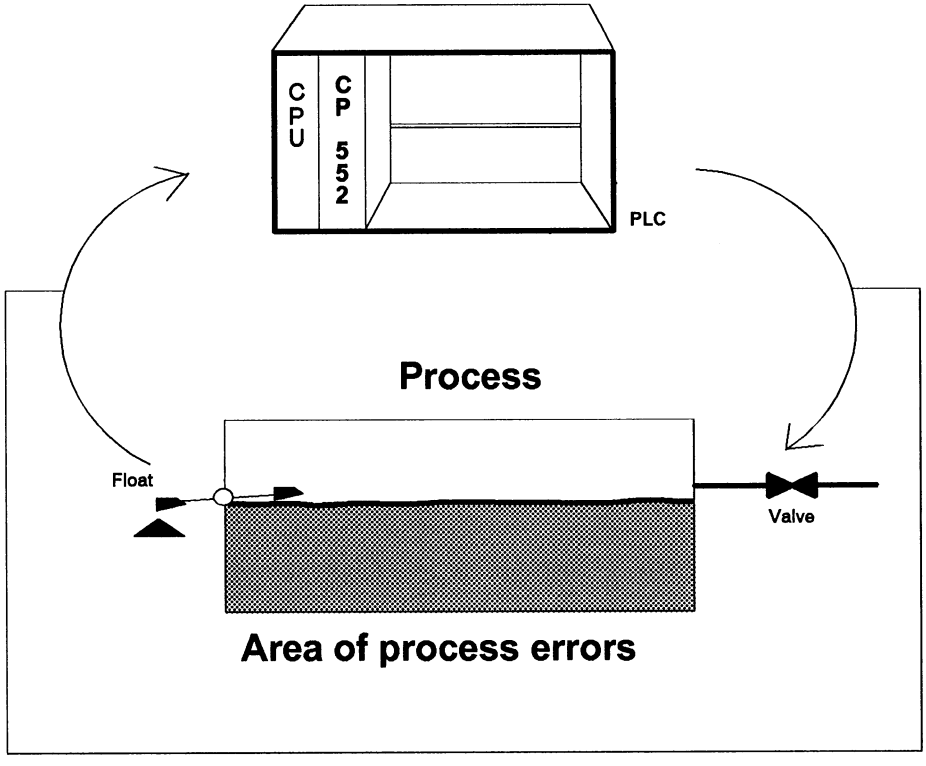


Fig. 1-2 Diagnosing process errors with the CP 552 - example of a process

You can use the CP 552 in the following programmable controllers of the SIMATIC® S5 U range:

S5-115U (only with external fan and adapter casing)
S5-135U (only with R processor or CPU 928)
S5-150U
S5-155U
EG 185 expansion unit

The CP 552 can only monitor **binary signals** (inputs, outputs and flags). It **cannot** monitor analog values, timers or counters.

The CP 552 recognizes process errors by comparing **setpoint data** and **actual data** belonging to the process.

- **Setpoint data** describe the **error-free process sequence**

  Using the programmer (PG), you create the setpoint data to be stored in the main memory of the CP 552.

- **Actual data** are the **current process image** (including flags).

  The central processor of the programmable controller (CPU) transfers the images of the inputs, outputs and flags to the CP 552 during each PLC cycle.

If the CP 552 detects discrepancies between the setpoint and actual data, it sends process error messages to the display unit.

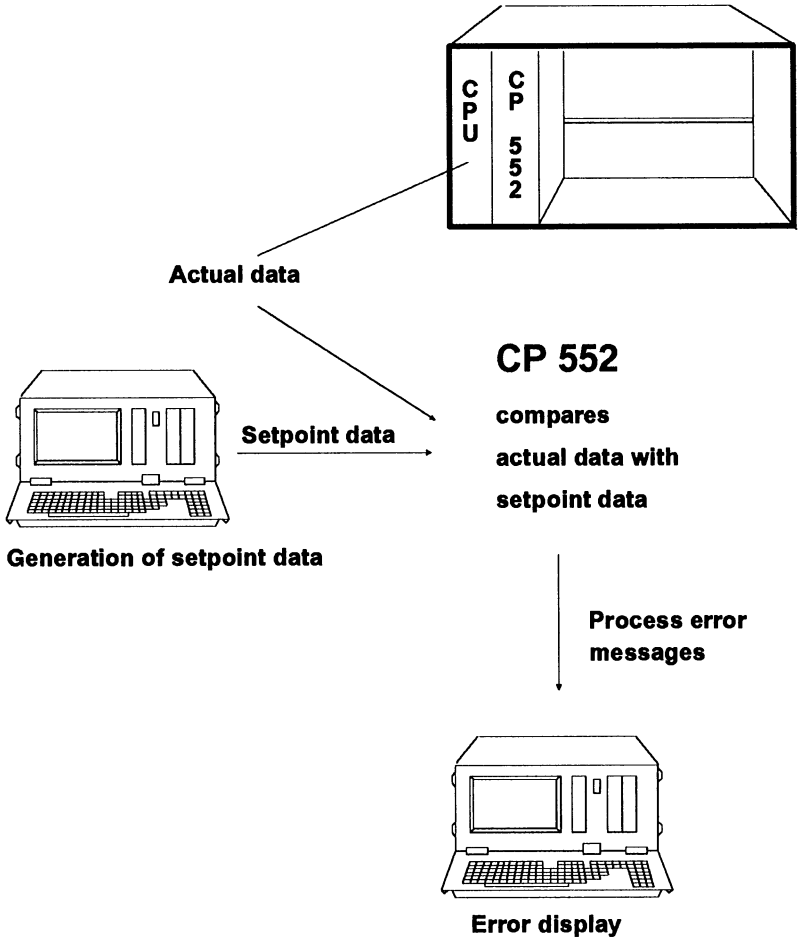


Fig. 1-3 System overview

Setpoint data are generated with the PG. The CP 552 has an AS 511 interface to which the PG is connected both for generating the setpoint data and for displaying errors. A monitor can also be connected directly to the VIDEO interface of the CP 552 for error display. The order numbers of the standard cable connectors to connect the CP 552/PG can be found in the ordering data.

The following blocks are required for the CPU of your PLC to allow data exchange between the CPU and the CP 552:

- **handling blocks and**
- **standard function blocks for process error diagnosis with the CP 552 diagnostic processor**

The software package DIAGNOSIS is available for process error diagnosis. Within the programming package **COM 552** you can perform the following:

- **generate setpoint data** (regardless of when your STEP® 5 user program was written)
- **transfer setpoint data to the CP 552**
- **specify the error displays**
- **display process error messages on the programmer**

Within the **LAD, CSF, STL** package you can also

- **generate setpoint data.**

This means that immediately after you enter a segment of your STEP® 5 user program, you can enter the corresponding setpoint data. The setpoint data can, however, only be transferred to the CP 552 in the programming package COM 552.

For further information, refer to the User's Guides "COM 552 Programming Package" and "LAD, CSF, STL Package with Process Error Diagnosis" in this manual. Order numbers can be found in the ordering data.

1.2 Displaying Process Control and System Messages

In addition to process error messages, the CP 552 can also display the following messages on the PG or on the local monitor:

- **System messages**

  These are messages about statuses or errors of the CP 552, generated by the CP 552 itself and transferred to the display unit. (Example of a system message: "CP 552 STOPPED.")

- **Process control messages**

  These are messages received by the CPU and sent to the CP 552 (e.g. messages from an IP 252). The CP 552 passes these messages to a display unit. Information about configuring process control messages can be found in this manual in the User's Guide "Displaying Process Control Messages" which is part of Volume 2.

Further information about these types of messages can be found in the User's Guide "Process Error Diagnosis with the CP 552."

1.3 Compatibility with Older Versions

All modules needed for effecting process error diagnosis (COM, CP, FBs) have been overhauled, in particular to make them easier to operate.

For this reason the "spare unit strategy" has been discontinued - it is no longer adopted. This has implications for you if you intend to use the existing, older programs and configurations with the new COM, CP, and FBs:

PLC program:

The parameter assignment of the FB 46 "SEND" has been modified. The FB call must be changed to match the new interface.

Message representation concept:

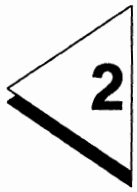
Having discontinued the spare unit strategy, the display concept (routing of the messages to the display units) must be adapted, if necessary, to the new demands.

Compatibility of the configuration files:

Configuration files generated with the old COM 552 version 04 (Order No. 6ES5998-3SE22) can be used further. However, specifications pertaining to the spare unit ID are ignored (for more information, refer to Volume 2/2, COM 552 Programming Package User's Guide, Unit ID in Part 4, Chapter 7).

It is therefore advisable to check the existing configurations.

Make sure when installing your applications that the module versions are compatible with each other. A mixture of old and new versions may result in malfunctioning.



The Two Versions of the CP 552

Contents

The CP 552 diagnostic processor is available in two versions, as follows:

CP 552-1:

This module consists of **one** board. You can store approximately **260 to 280 setpoint data elements** in the CP 552-1. This number corresponds approximately to the same number of segments. There are approximately 55 Kbytes available for setpoint data elements in the memory of the CP 552-1. The term "setpoint data element" is explained in the User's Guide "Process Error Diagnosis with the CP 552."

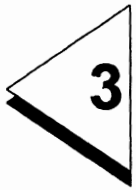
CP 552-2:

This module consists of **two** boards. The CP 552-2 has a **640 Kbyte buffered CMOS-RAM submodule.** Because of the RAM module, you can store far more data on the CP 552-2 than on the CP 552-1. The CP 552-2 has approximately 695 Kbytes available for setpoint data elements.

To **display on the local monitor** you must transfer symbols and process control messages to the CP 552. This leaves less space for the setpoint data elements. Space for **1** setpoint data element is lost when 3 process control message texts or 12 symbols are transferred.

Rule of thumb for the length of a setpoint data element:
51 bytes + comment length + 16 (number of terms)

The order numbers of the modules can be found in the ordering data in this manual. The CP 552-1 **cannot** be upgraded to the CP 552-2.



Design of the CP 552

Contents

3.1 Description of the Front Panel

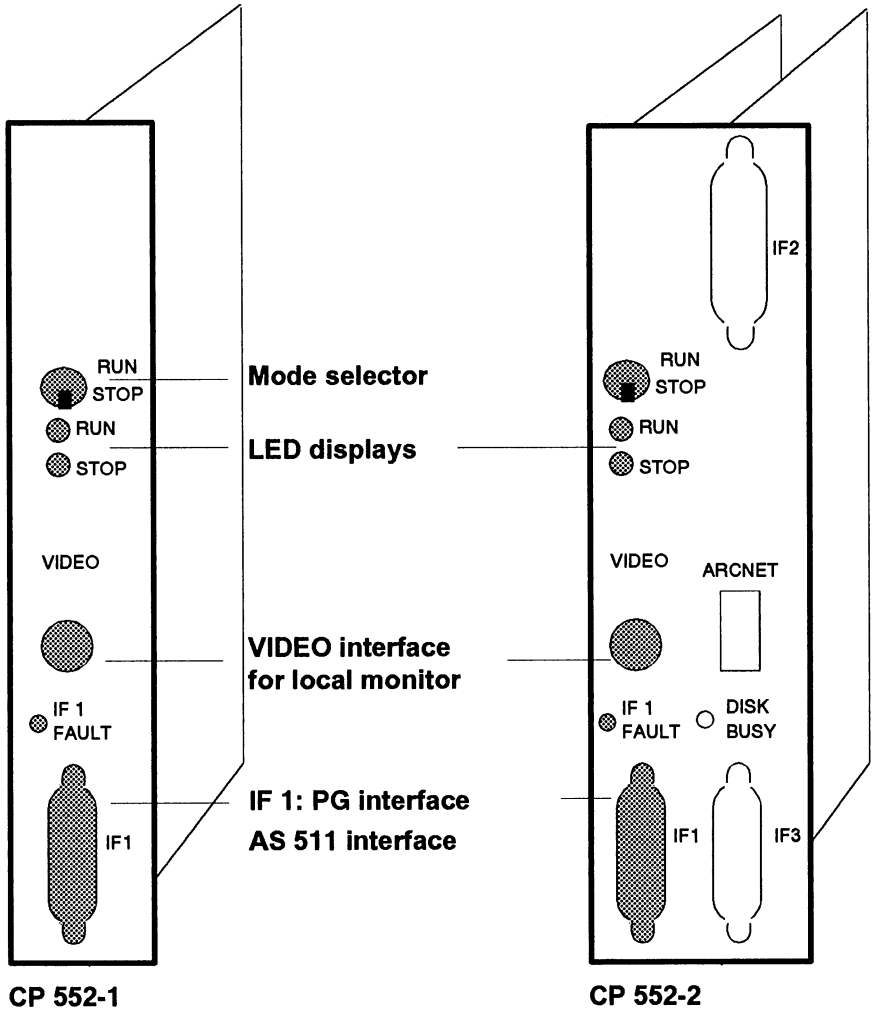


Fig. 3-1 Front panels of the CP 552-1 and CP 552-2
Ignore the white elements for the present time

The front panel of the CP 552 has the following **controls and displays**:

Mode selector:

RUN: the processor of the CP 552 executes the programmed functions
STOP: the processor is in the STOP state

LED displays:

LED "RUN" lit, green:

- the mode selector is set to RUN, there is a connection to the CPU via the S5 bus and the processor executes the programmed functions.

LED "STOP" lit, red:

- the mode selector is set to STOP or
- the mode selector is set to RUN, however, the CP 552 is not yet synchronized with the CPU (function or data blocks missing or assigned incorrect parameters) or
- the mode selector is set to RUN, the processor is, however, in the STOP state, and no cold restart has been performed at the PG or
- the CP 552 is still in the start-up phase following a power failure or
- a serious error was detected during the start-up.

LED "IF 1 FAULT" lit, red:

- the self-test has detected a firmware error (EPROM check).

3.2 Connection to the PG

The programmer is connected to the **PG interface IF1** (AS 511 interface). There are standard cable connectors available for this connection (the order numbers can be found in the ordering data). The device interface operates in a serial, asynchronous mode with current loop (TTY) signals.

In the S5-135U with the KOR C coordinator and in the EG 185, the PG interface can also be addressed via the S5 bus.

3.3 Connection to the S5 Bus

For the connection to the S5 bus in the programmable controller

- the CP 552-1 has **two backplane connectors**
- the CP 552-2 has **three backplane connectors**

3.4 Connection to the Local Monitor

You can connect a local monitor (black and white) to the **VIDEO interface**. A suitable cable is supplied with the monitor (coaxial cable acc. to DIN 47295). Further information regarding the technical data can be found in Section 3.7.

3.5 Dual-port RAM: Interface to the Central Processor

The data exchange between the CP 552 and the CPU of the PLC uses a common memory area, the **dual-port RAM**. The dual-port RAM is located on the CP 552 and can be compared with a mailbox. The CP 552 and the CPU can deposit information (data, requests etc.) for each other in this mailbox.

The CPU has the initiative during data exchange. The CP 552 must be "asked" by the CPU whether or not it has data to transfer. This function and the transfer of the actual data from the CPU to the CP 552 is performed by the following standard function blocks (for order numbers, see ordering data):

- **Handling blocks**
- **Standard function blocks for process error diagnosis with the CP 552 diagnostic processor**

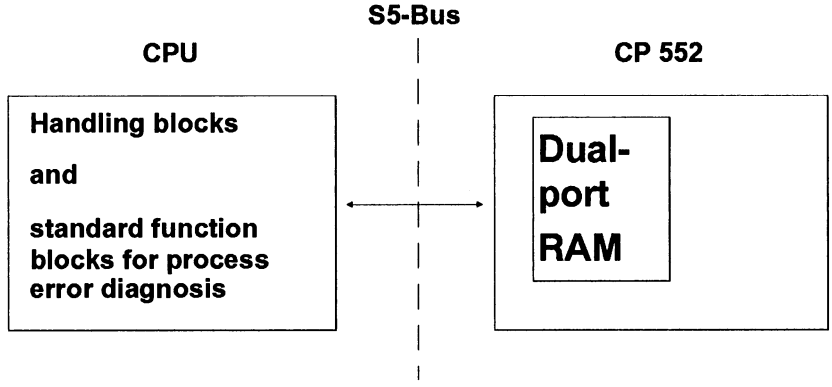


Fig. 3-2 Dual-port RAM: interface between the CP 552 and CPU

Handling blocks and standard function blocks for process error diagnosis organize the data exchange and transfer the actual data.

Data transfer area: pages

The areas within the dual-port RAM in which the exchange of data is organized and in which data is transferred, are known as pages. The CP 552 has two pages, each 1 Kbyte long. The pages of all the CPs in a PLC are in the same address area. Each page has an **interface number** assigned to it. This is **between 0 and 255**.

Each CP has a **page select register**. The page select registers of all the CPs in a PLC are at the same address.

If the CPU wishes to transfer data to a CP, it enters the interface number of the required page in all the page select registers available. Each CP compares the content of its page select register with its set interface numbers. The CP 552 only releases a page for data transfer when the contents of its page select register match one of its interface numbers. Communication with all other pages and the S5 bus is then blocked.

You must set the interface number (SSNR) of your CP 552. For further information, refer to Section 6.1 "Setting the Interface Number." To simplify the setting of the interface numbers, you only need to set the interface number of one page on your CP 552. This page must be assigned an **even interface number** - the other page then has the next higher odd interface number.

Interface numbers must only be assigned once.

3.6 Hardware Clock

The CP 552 has a hardware clock buffered by the PLC. You can display the date and time.

The CP 552 software automatically takes into account both the change of year and leap years. This also applies when the unit is switched off and in the buffered state (up to one year).

3.7 Technical Data [1]

Type of protection:	IP 00
Permitted ambient temperature:	0 °C to 55 °C (+32 to +131 °F)
Transport and storage temperature:	-40 °C to +70 °C (-56 to +186 °F)
Humidity rating acc. to DIN 40040:	Code letter F
Operating altitude:	up to 3500 m above sea level
Weight:	approx. 0.4 kg with one board approx. 0.9 kg with two boards
Power supply:	5 V, tolerance max. 5% CP 552-1: max. 1.8 A CP 552-2: max. 3.2 A
Design:	CP 552-1: dimensions 160 mm x 233.4 mm front panel width approx. 20 mm 2 ES 902, row 2 backplane connectors 2 x 48-pin 1 15-pin Cannon front connector 1 BNC connector CP 552-2: dimensions 160 mm x 233.4 mm front panel width approx. 40 mm 3 ES 902, row 2 backplane connectors 2 x 48-pin, 1 x 8-pin 2 15-pin Cannon front connectors 1 25-pin Cannon front connector 1 BNC connector 1 light guide connector

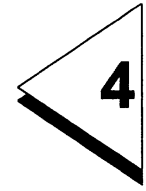
[1] *For multiples of two the following conventions have been used in this manual: K*byte=2^{10}; M*byte=2^{20}; G*byte=2^{30}*

Logical data:

Microprocessor:	80186 from INTEL; 15 MHz quartz frequency
Memory configuration:	
Operating system EPROM	128 K*bytes
CMOS RAM (CP 552-1 and CP 552-2)	128 K*bytes
Expansion: CMOS-RAM submodule (only CP 552-2)	640 K*bytes
Dual-port RAM	2 K*bytes (2 pages)
Character generator	32 K*bytes
Serial PG interface	TTY max. 1000 m
Transmission rate to PG	9600 bps

VIDEO interface for connecting the local monitor:

Black and white CRT controller	ASCII characters
Principle	50 Hz
Image refresh frequency	15 kHz
Line frequency	25 lines x 80 characters
Raster field per character	7 x 11
Video signal	composite signal



Assembly

Contents

4.1 Inserting and Removing the CP 552

The CP 552 must only be inserted or removed when the 5 V and 24 V power supply is switched off!

Some of the PLC racks have metal springs in the guide rails which could get bent when the CP 552 is inserted. Ensure that the CP 552 is correctly inserted.

4.2 Slots in the Programmable Controller

The CP 552-1 occupies **one slot in the programmable controller**.

The CP 552-2 occupies **two slots**. Remember that only the left-hand board must be inserted in the "CP" slot, the right-hand board is then automatically in the slot to the right and must simply be supplied with 0 V and 5 V.

The CP 552-1 and CP 552-2 can be inserted in the following slots:

S5-135U

(old rack, Order No. 6ES5 135 - 3KAXX)

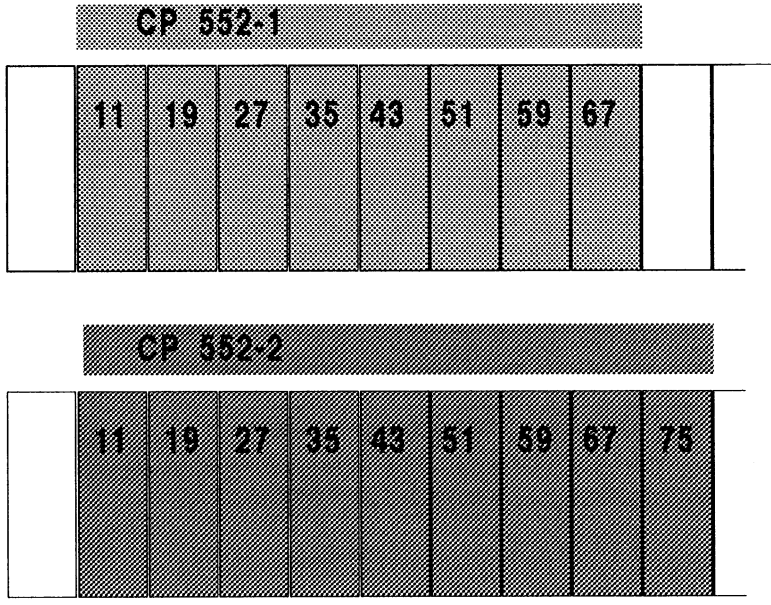


Fig. 4-1 Slots in the S5-135U

CP 552-1 occupies one slot.
CP 552-2 occupies two slots, both boards must be inserted within the marked area.

S5-135U

(new rack, Order No. 6ES5 135 - 3UAXX)

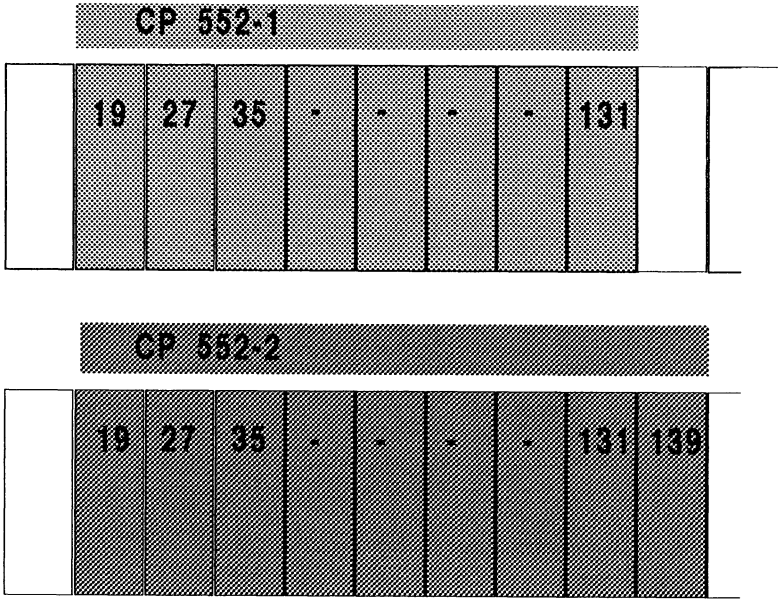


Fig. 4-2 Slots in the S5-135U

CP 552-1 occupies one slot.
CP 552-2 occupies two slots, both boards must be inserted within the marked area.

S5-150U

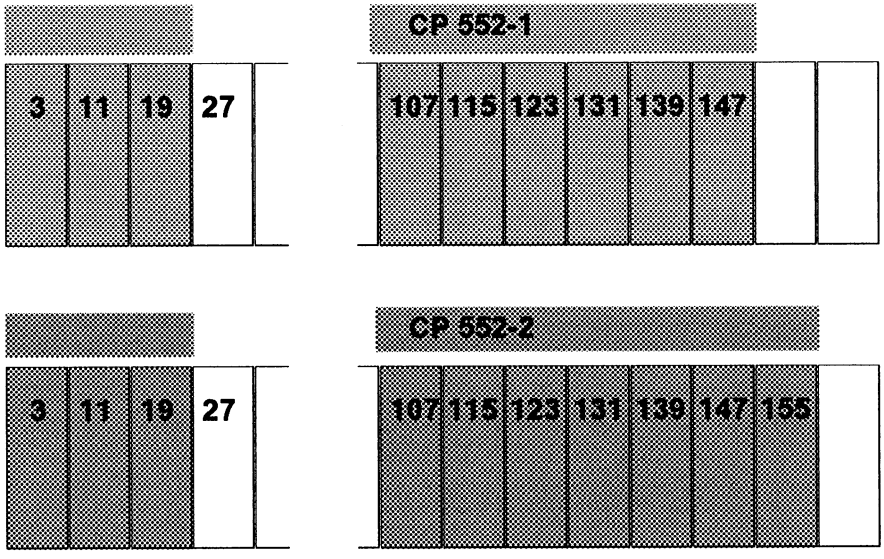


Fig. 4-3 **Slots in the S5-150U**

CP 552-1 occupies one slot.
CP 552-2 occupies two slots, both boards must be within the marked area.

If you insert CPs in slots 3, 11 or 19, you must insert a matrix module in slot 27.

If you use slots 139 and 147, jumpers 1 to 8 must be inserted on the bus board.

S5-155U

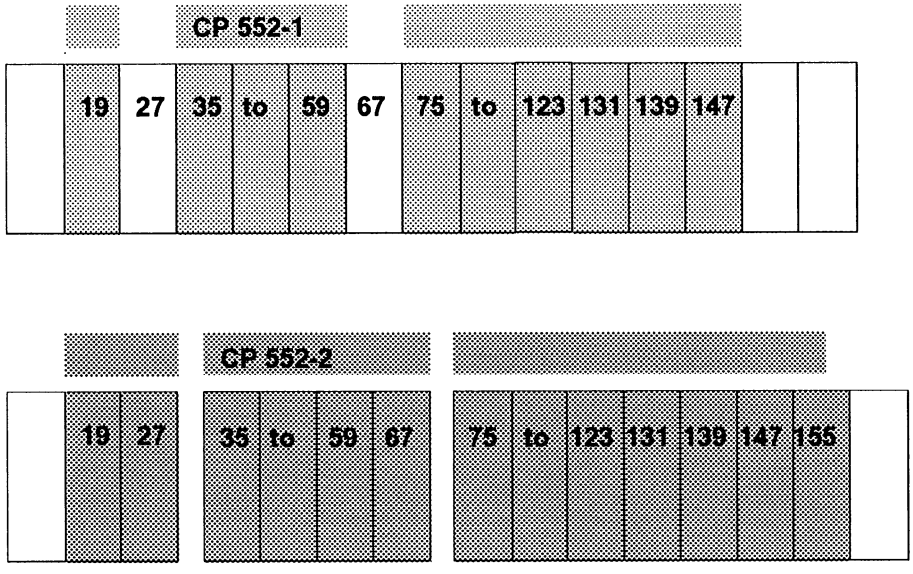


Fig. 4-4 **Slots in the S5-155U**

CP 552-1 occupies one slot.
CP 552-2 occupies two slots, both boards must be within the marked area.
If you use slots 139 and 147, you must change the jumpers on the bus board.

EG 185U

Interface modules 304 and 314 required

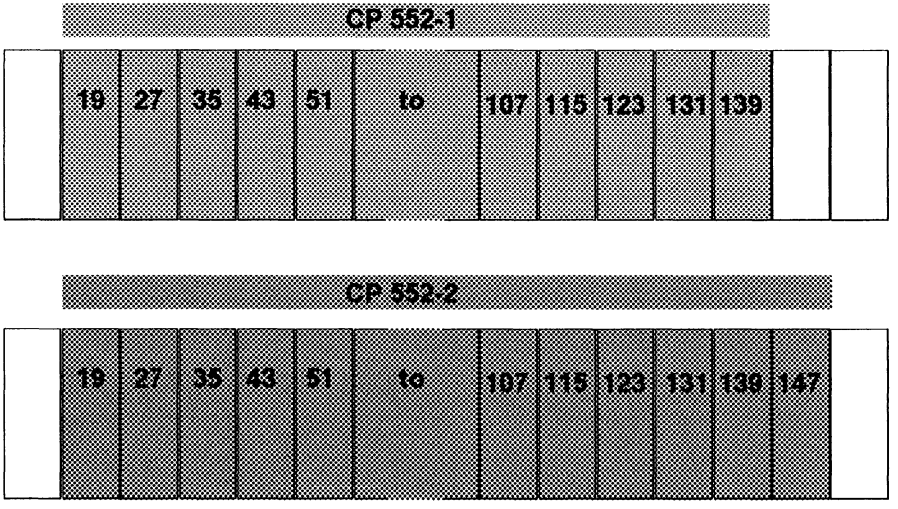


Fig. 4-5 Slots in the EG 185U

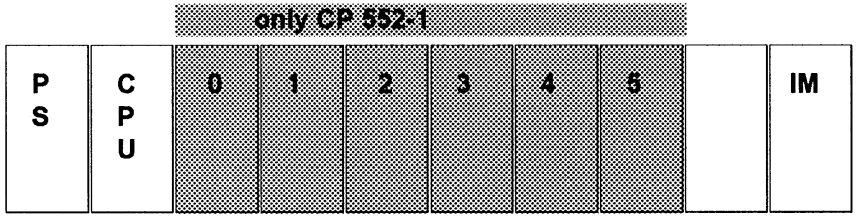
CP 552-1 occupies one slot.
CP 552-2 occupies two slots, both boards must be within the marked area.

S5-115U

Can only be used with the adapter casing

Subrack CR 700-2:

Fan required



Subrack ER 701-3:

Fan required; interface modules 304 and 314 required

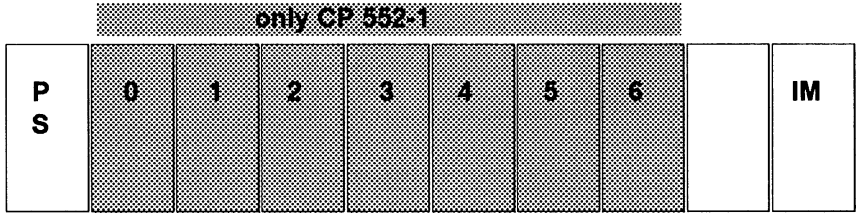


Fig. 4-6 Slots in the S5-115U
CP 552-1 occupies one slot.

S5-115U

Can only be used with the adapter casing

Subrack CR 700-3:

Fan required; interface modules 304 and 314 required

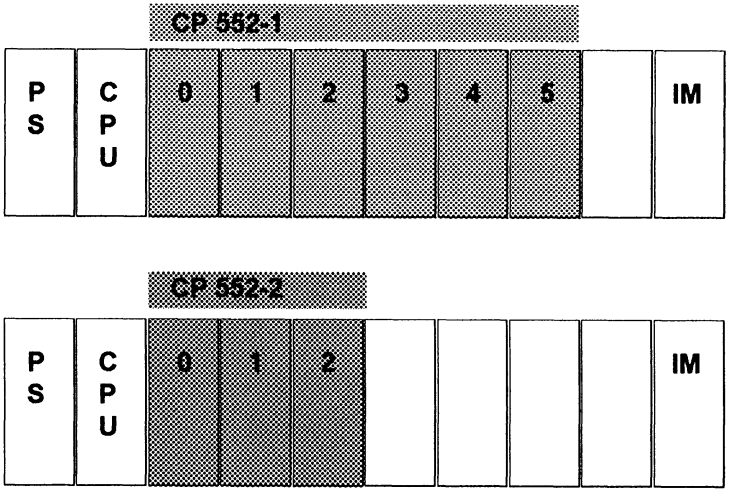


Fig. 4-7 Slots in the S5-115U

CP 552-1 occupies one slot.
CP 552-2 also occupies one slot.

For further information about your PLC, refer to the appropriate manual.

4.3 Installation Guidelines

The module should be installed according to the installation guidelines in the manual supplied with your programmable controller. The connecting cable between the CP 552 and programmer must have a grounded shield at both ends. This requirement is met by the standard cable connectors listed in the catalog (see ordering data).

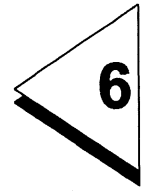
Installing the CP 552 for the First Time

Contents

When using the CP 552 for the first time, work through the following checklist:

- Check the jumper settings (Chapter 6) and set the interface number (Section 6.1).
- Insert the CP 552 module in a suitable slot in the programmable controller (see Chapter 4). Make sure that the programmable controller is switched off when you do this.
- Check the mode selector on the front of the CP 552. This must be switched to "STOP."
- You can now switch on the power supply to your programmable controller. The operating system runs through a self-test. If a fault is detected, the red LED "IF 1 FAULT" is lit.
- Now generate the setpoint data for your STEP® 5 user program on the programmer and transfer this to the CP 552. For further information, refer to the User's Guides "COM 552 Programming Package" and "LAD, CSF, STL Package with Process Error Diagnosis" in this manual.
- Carry out a cold restart on the CP 552: switch the mode selector to "RUN."
- Program the error display on the programmer (see User's Guides "COM 552 Programming Package" and "Error Display on the Local Monitor"). You can now begin process error diagnosis.

Jumper Settings



Contents

The CP 552-1 and CP 552-2 have the following jumpers and switches:

- **fixed jumpers and switches**

  These are correctly set in the factory and must not be changed.

- **DIP switch J56, which is used to set the interface number (see Section 6.1).**

Note on Figs. 6-1, 6-2, 6-3, 6-4, 6-5 and 6-6 on the following pages:

The CP 552 has different types of jumpers (e.g. double jumpers and triple jumpers). "X..." is used to designate the jumpers. Which jumpers are inserted and which are not inserted when the modules are supplied can be seen on the following pages. "J..." identifies a DIP switch or a mounting location.

CP 552-1 side view

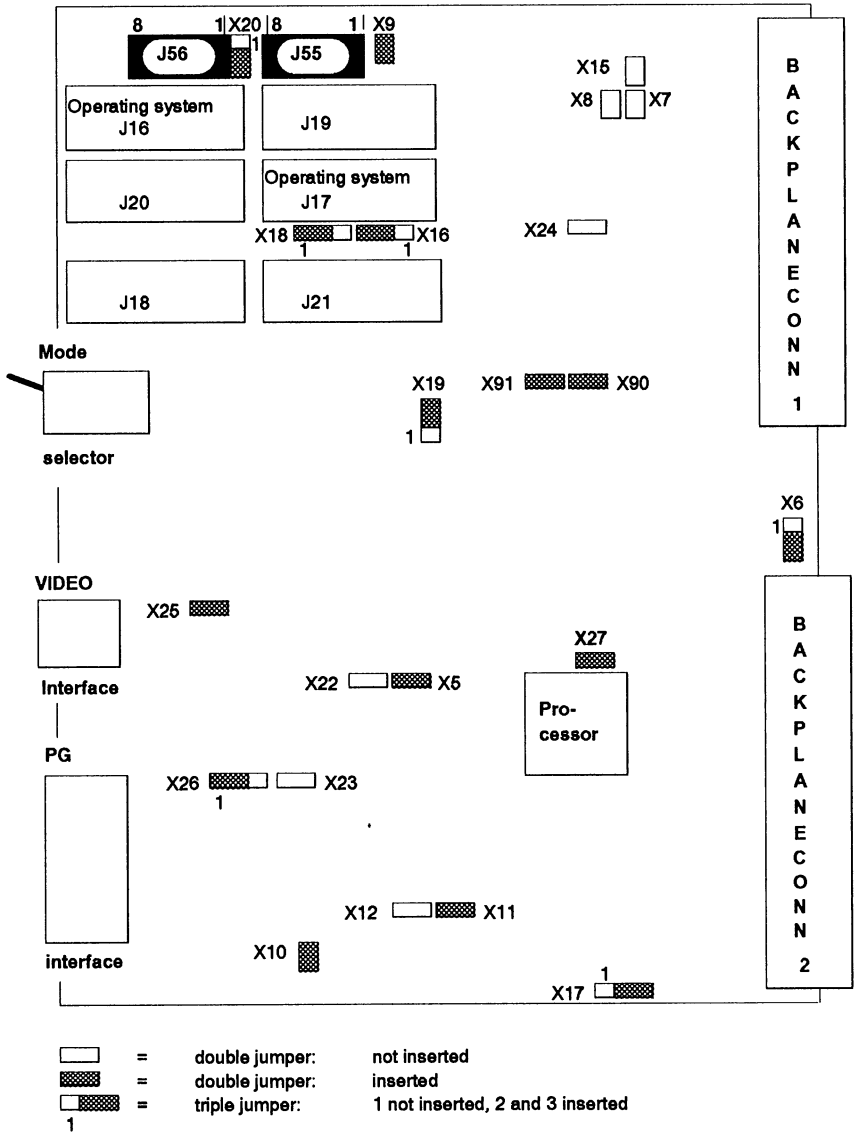


Fig. 6-1 Jumper settings as supplied

CP 552-1 viewed from above

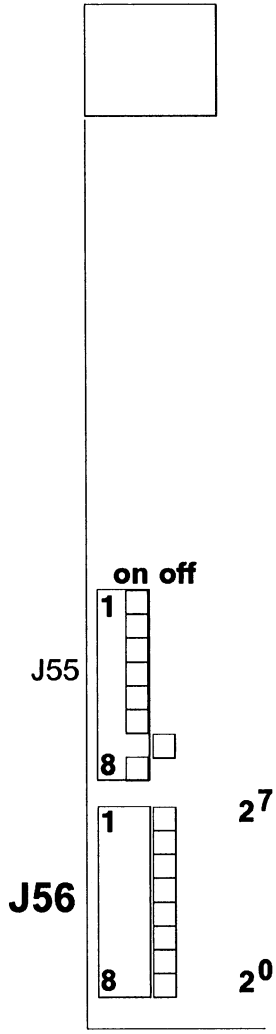


Fig. 6-2 Jumper settings as supplied

The interface number is set using DIP switch **J56** (see Section 6.1).

J55 sets the base address. When the module is supplied, this is set to 61 Kbytes (F400H) for the use of handling blocks. This setting should not be changed.

	Value
J55 /1 = not used	-
J55 /2 = not used	-
J55 /3 = on	2^5
J55 /4 = on	2^4
J55 /5 = on	2^3
J55 /6 = on	2^2
J55 /7 = off	2^1
J55 /8 = on	2^0

The base address 61 Kbytes is calculated as follows:

$2^0 + 2^2 + 2^3 + 2^4 + 2^5 = 1 + 4 + 8 + 16 + 32 = 61$

CP 552-2 side view of the left-hand board

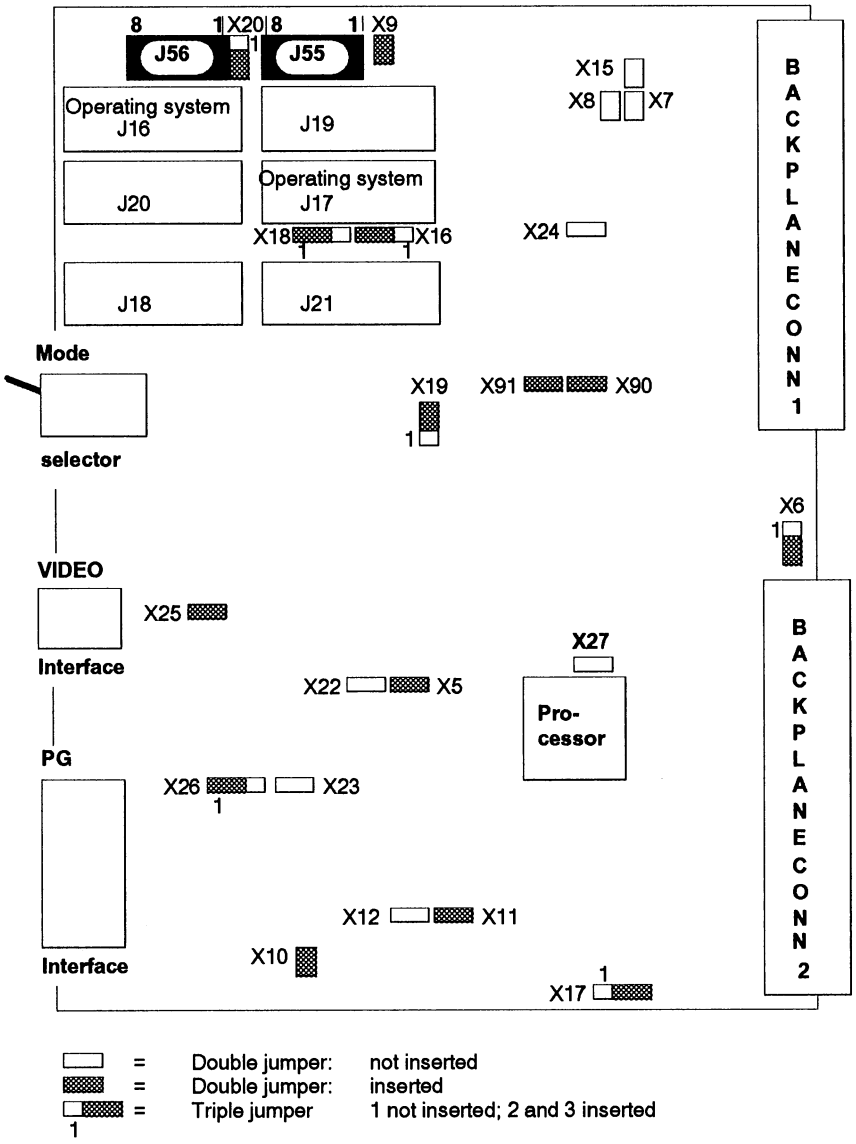


Fig. 6-3 Jumper settings as supplied

CP 552-2 side view of the right board with the CMOS-RAM submodule inserted

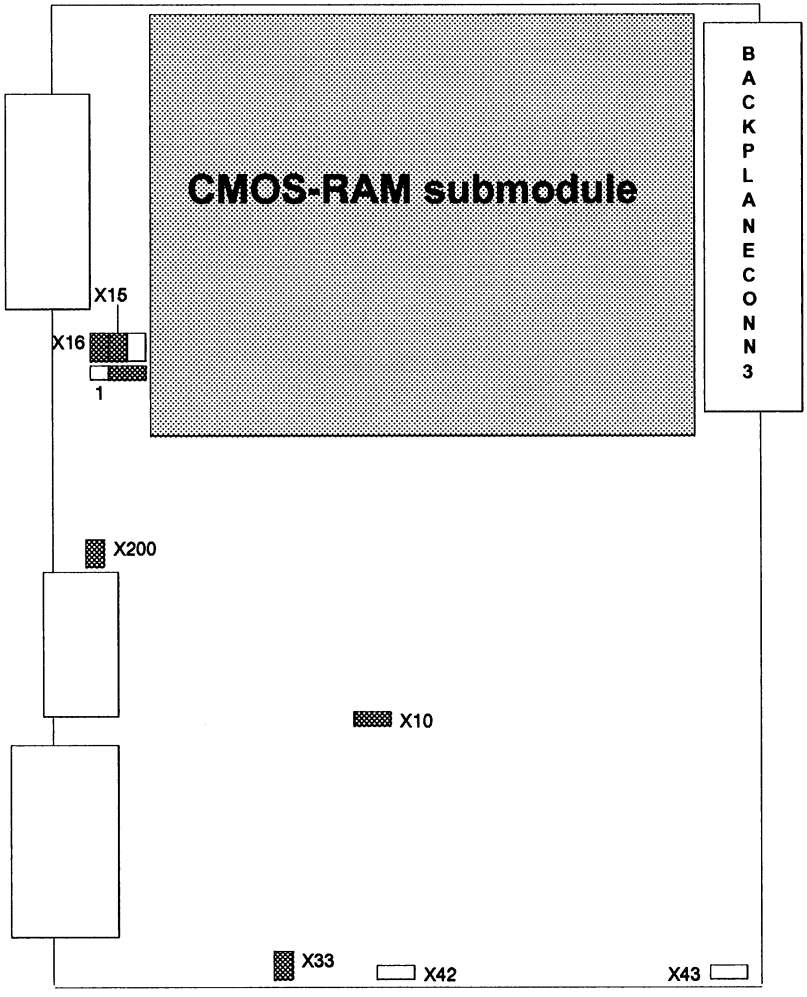


Fig. 6-4 Jumper settings as supplied

CP 552-2 side view of the right board without CMOS-RAM submodule

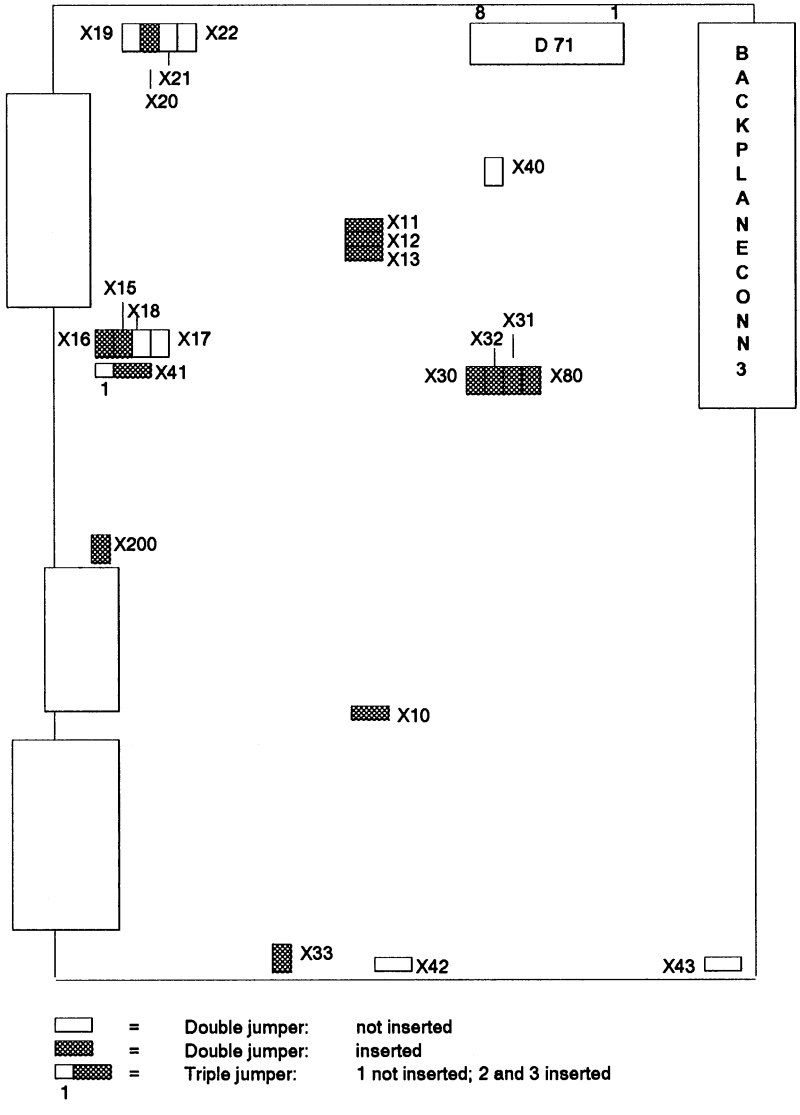


Fig. 6-5 Jumper settings as supplied

CP 552-2 viewed from above with inserted CMOS-RAM submodule

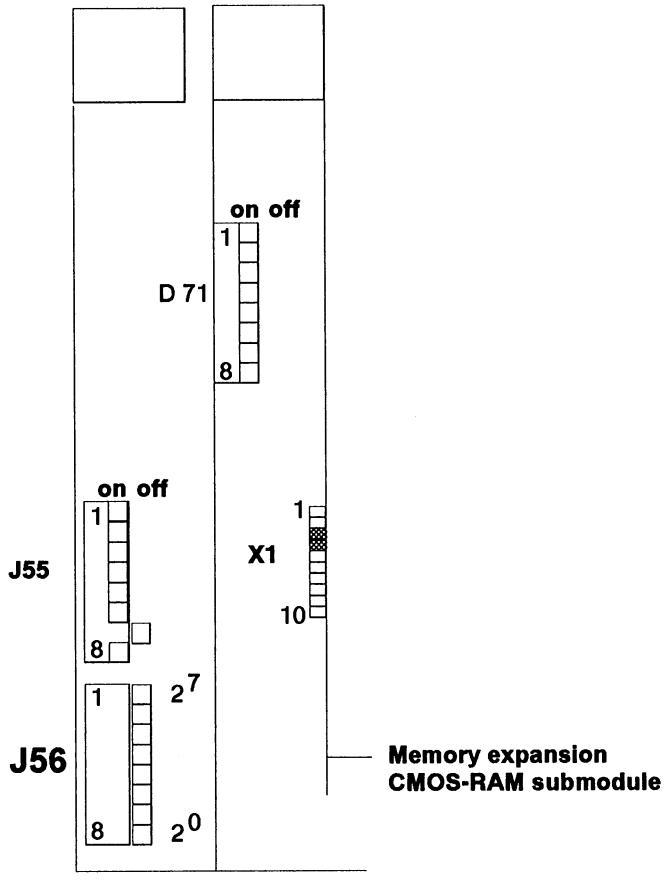


Fig. 6-6 Jumper settings as supplied

The DIP switch **J56** is used to set the interface number (see Section 6.1).

J55 sets the base address. When the module is supplied, this is set to 61 Kbytes (F400H) for the use of handling blocks. You should not change this setting.

	Value
J55 /1 = not used	-
J55 /2 = not used	-
J55 /3 = on	2^5
J55 /4 = on	2^4
J55 /5 = on	2^3
J55 /6 = on	2^2
J55 /7 = off	2^1
J55 /8 = on	2^0

The base address 61Kbytes is calculated as follows:

$2^0 + 2^2 + 2^3 + 2^4 + 2^5 = 1 + 4 + 8 + 16 + 32 = 61$

X1 on the memory expansion sets the address area of the CMOS-RAM submodule: 256 Kbytes to 896 Kbytes, when pins 3 and 4 are connected (see Fig. 6-6).

6.1 Setting the Interface Number

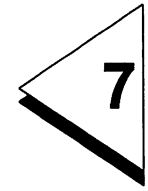
The interface number is set at DIP switch **J56** (see Fig. 6-2 and 6-6). Using the interface number, the handling blocks can address the CP 552. For further information, refer to Section 3.4. You can set 256 different combinations as follows:

	Value
J56 /1	2^7
J56 /2	2^6
J56 /3	2^5
J56 /4	2^4
J56 /5	2^3
J56 /6	2^2
J56 /7	2^1
J56 /8	2^0 not relevant

You can only set the **even interface number** for the first page of the CP 552. The next odd interface number of the second page of the CP 552 is then set automatically.

Settings as supplied

All the switches are off, i.e. the first page of the CP 552 is assigned **interface number 0**, the second page automatically has **interface number 1**.



Pin Assignments

Contents

7.1 Backplane Connector 1

	d	b	z
2		Ground	+5 V
4	U BAT	PESP	
6	ADB12	ADB 0	*
8	ADB13	ADB 1	/MEMR
10	ADB14	ADB 2	/MEMW
12	ADB15	ADB 3	/RDY
14	/IRA	ADB 4	DB 0
16		ADB 5	DB 1
18		ADB 6	DB 2
20		ADB 7	DB 3
22		ADB 8	DB 4
24		ADB 9	DB 5
26		ADB 10	DB 6
28	/DSI	ADB 11	DB 7
30		BASP	
32	/BASPq	Ground	/HALT

Backplane connector 1

* On 6ES5 552-3UA11 as from Version 06 and 6ES5 552-UA21 as from Version 05 this pin is no longer assigned the CPKL signal.

7.2 Backplane Connector 2

	d	b	z
2		Ground	+5 V
4			
6			
8			
10			
12			
14			/NAU
16			/BAU
18			
20		/STEU	
22	TXD	/STOPPq	
24			GEP
26		/RXD	
28		/PERO	
30			M 2
32		Ground	+24 V

Backplane connector 2

7.3 Backplane Connector 3

Only the CP 552-2 has this backplane connector.

	d	b	z
2		Ground	+5 V
4			
6			
8			
10			
12			
14			
16			
18			
20			
22			
24			
26			
28			
30			
32		Ground	

Backplane connector 3

7.4 PG Interface (IF 1)

Pin	Designation
1	Ground (Mext)
2	Receiver TTY (-)
3	not used
4	+24 V
5	Connector identifier (keyboard, PG) TTL level
6	Transmitter TTY (+)
7	Transmitter TTY (-)
8	Ground (Mext)
9	Receiver TTY (+)
10	24 V ground (current sources (-) 20 mA)
11	Current source (+) 20 mA
12	0 V
13	Current source (+) 20 mA
14	not used
15	not used

Index

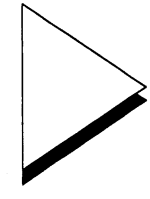
SIEMENS

SIMATIC S5

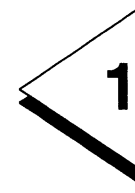
Process Error Diagnosis with the CP 552

User's Guide

C79000-B8576-C670-05



Contents



System Overview

Contents

1.1 Terms

As an introduction to the User's Guide, this section discusses certain terms required in conjunction with process error diagnosis. Some of these terms will already be known to you from process engineering, whereas others will be new.

A **process** consists of several **process elements**. A process element is a self-contained activity. This activity can only be executed when certain requirements are met. The activity leads to a defined final status. An example of a process element can be seen in Fig. 1-2, in which a motor-driven carriage must reach the right-hand limit switch within a certain time.

Process errors are errors or faults which occur outside the programmable controller, for example, defective plant equipment, sensors, actuators, wire breaks.

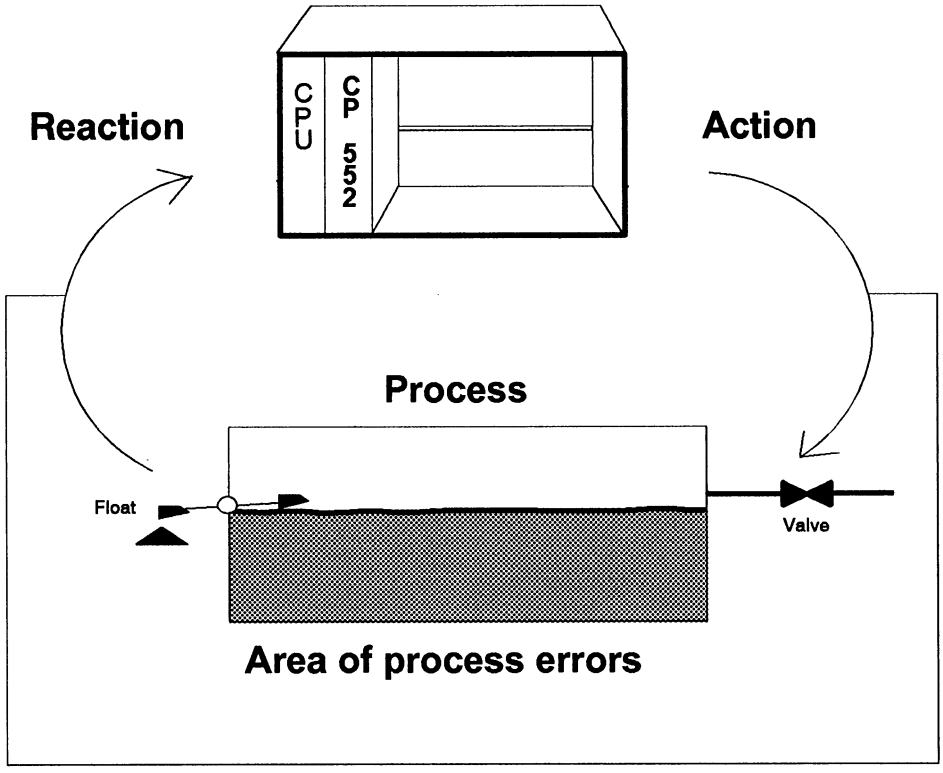


Fig. 1-1 Diagnosing process errors with the CP 552 diagnostic processor - example of a process

The **action** is the trigger, which the process receives from the controller.

The **reaction** is the response of the process.

Example for Fig. 1-1:

Action:	Q 1.0 = Valve open
Reaction:	I 5.7 = Tank full

The **setpoint data** describe the error-free execution of the process by means of a number of binary signals and monitoring times. The more signals that are available, the better the quality of the process error diagnosis will be.

The setpoint data consist of individual **setpoint data elements**.

A **binary signal** can be an input (I), output (Q) or flag (F).

The **actual data** is the current process image including flags.

The **release** releases a setpoint data element. This must be present the whole time that the setpoint data element is monitored. For this reason, only static signals (signal level) can be used.

If an action leads to a defined reaction, this reaction can serve as the **start condition** to reactivate the whole process.

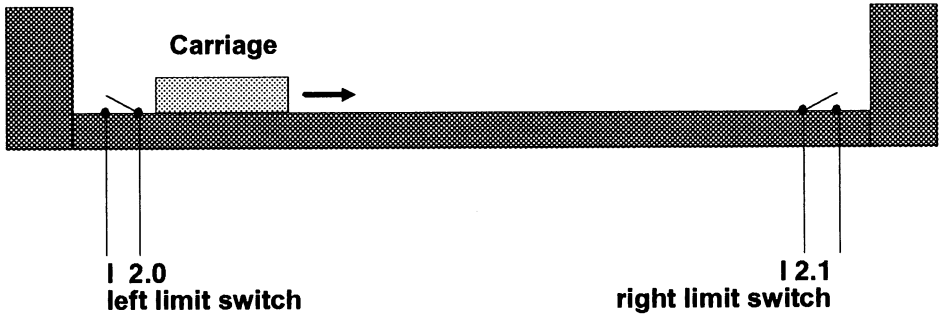


Fig. 1-2 Example "carriage"

The start condition to start this equipment is as follows: the carriage is in the left start position. The start position will be monitored if you enter "left limit switch activated" in a setpoint data element as reaction. When the equipment is started up, a check is made to determine whether this reaction is present or not. If the reaction is not present, the CP 552 outputs an error message.

An action results from the logical combination of several signals. Within these signals, the distinction is made between triggers and interlocks.

The **trigger** is the actual initiator of the action, the **interlocks** provide the necessary process environment.

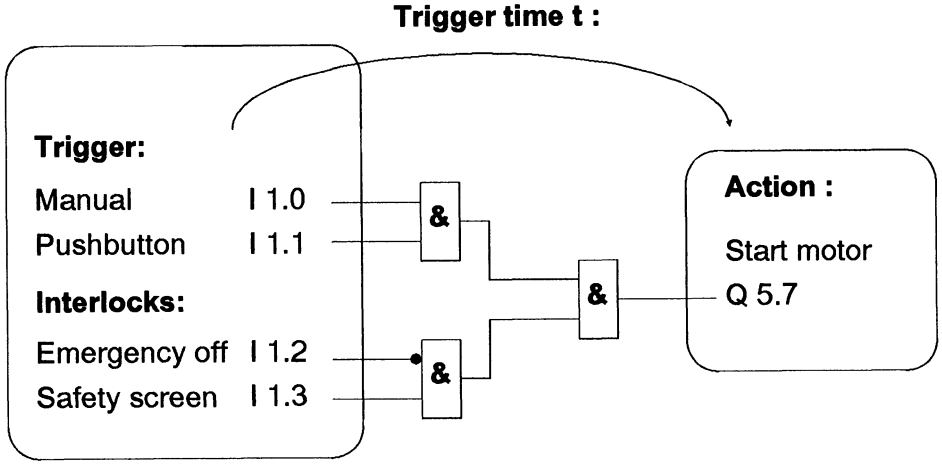
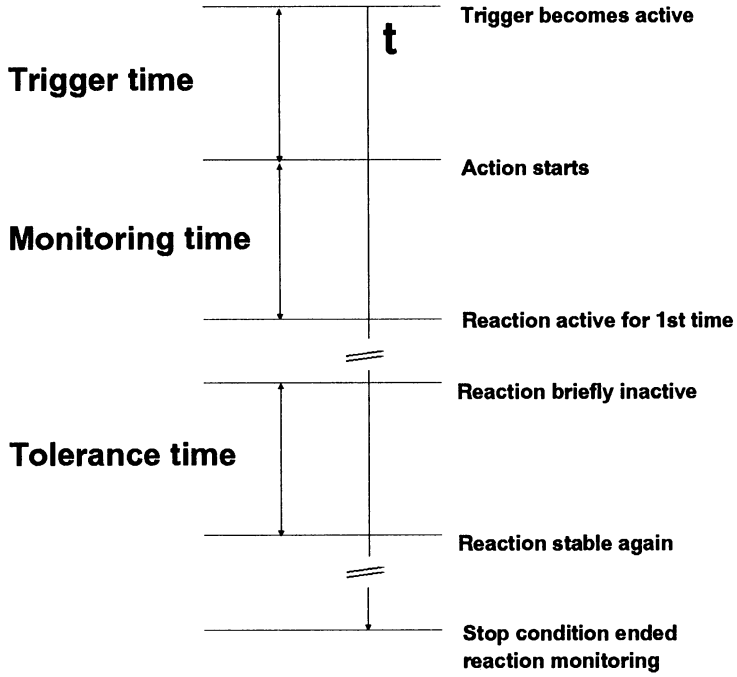


Fig. 1-3 Example - trigger and interlocks

The **trigger time, monitoring time** and **tolerance time** monitor the course of certain process functions.

The trigger time begins when the trigger becomes active and is completed when the action starts. The monitoring time begins when the action starts and is completed when the reaction becomes active. The tolerance time begins when the reaction is no longer active and is completed when the reaction becomes active again.



The monitoring of a process element is stoppped when a declared **stop condition** is met.

The trigger, interlocks, action, reaction and stop condition are logical expressions, consisting of terms. They include the following:

- signal levels or
- signal edges or
- signal levels and edges.

Within a **term** a binary signal is assigned to a signal level or signal edge. Terms can be ANDed or ORed with each other. The AND before OR rule applies as in STEP® 5. **Bracketed levels are not allowed.**

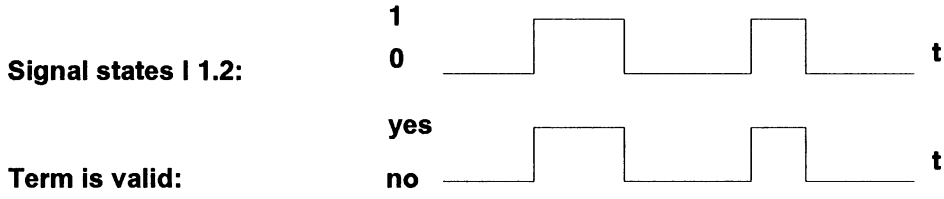
The **signal level** can be "0" or "1."

The **edge** can be positive-going ("p" = 0/1 transition) or negative-going ("n" = 1/0 transition).

Examples of terms

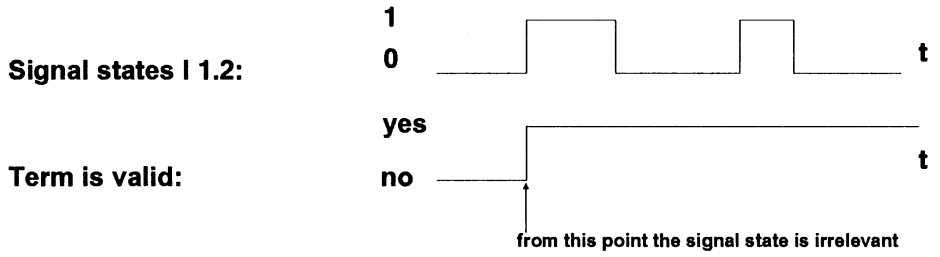
A binary signal is assigned to a **signal level**: I 1.2 = 1

The term is then valid, when signal I 1.2 has the signal level 1 when it is scanned.



A binary signal is assigned to an **edge**: I 1.2 = p

The term is valid when the signal I 1.2 has changed from 0 to 1 when it is scanned.



Edges are only stored within an AND block when all previous terms within the AND block are valid.

Example 1:

```
        Manual operation = 1
(A)     Pushbutton = P
```

The positive edge of the pushbutton is only stored when manual operation is active.

Example 2:

```
        I 1.0 =1
(A)     I 2.1 = P ; edge is only stored if I 1.0 = 1
(A)     F 4.1 =N ; edge is only stored if I 1.0 = 1 and I 2.1 = 1
(O)     I 12.4 = 1
(A)     I 4.2 = P ; edge is only stored if I 12.4 = 1
(O)     Q 5.4 = N ; edge is always stored.
```

The following points should be remembered in an expression using edges:

- The edges must occur while the setpoint data element is being monitored.
- The edges are cleared when the setpoint data element is fulfilled, e.g. the stop condition becomes active.

1.2 Functions of Process Error Diagnosis

As shown in Fig. 1-4, you can subdivide a process. Process error diagnosis with the CP 552 diagnostic processor can monitor both dynamic functions and statuses and detect errors or faults.

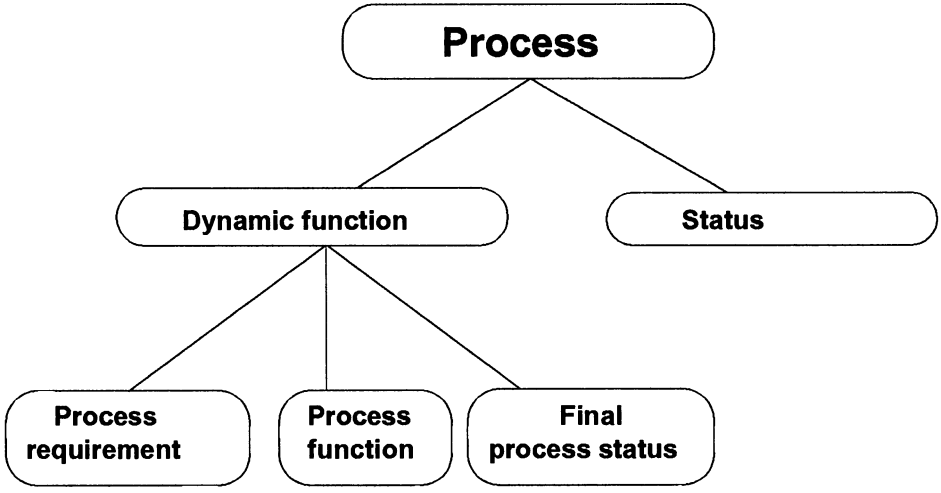


Fig. 1-4 Functions of process error diagnosis

Example of monitoring a dynamic process (Fig. 1-5):

A motor-driven carriage must reach the right-hand limit switch within a certain time.

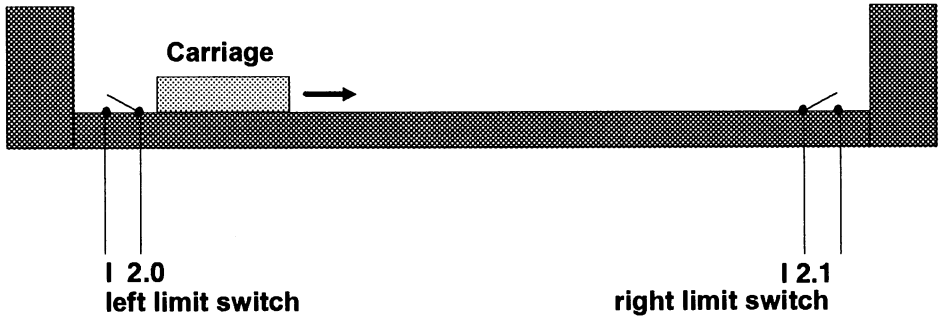


Fig. 1-5 Monitoring a dynamic function and a status

Example of monitoring a status (Fig. 1-5):

The left and right limit switches must not be active simultaneously (monitoring pairs of limit switches).

Note:

Where **monitoring involves safety**, you must also include appropriate **interlocks in the STEP® 5 user program** for process error diagnosis with the CP 552. In the worst case, errors may be signalled with a delay up to 1 second, or error messages may be lost if there is a CP 552 buffer overflow.

1.2.1 Monitoring Dynamic Functions

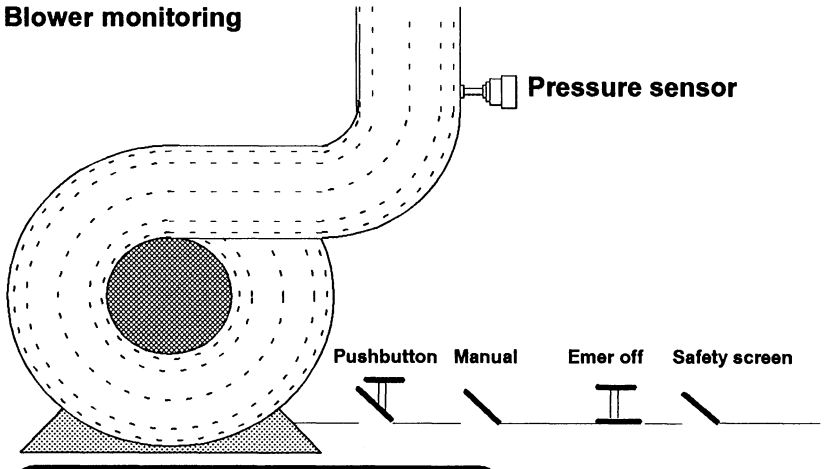
There are different types of diagnosis, as follows:

- **Interlock diagnosis**: monitoring the process requirement
- **Action diagnosis:** monitoring the process function
- **Reaction diagnosis**: monitoring the final process status

To monitor a process element, you can select one of the following options:

- **Interlock diagnosis** or
- **Action diagnosis** or
- **Interlock diagnosis and action diagnosis** or
- **Action diagnosis and reaction diagnosis** or
- **Interlock diagnosis, action diagnosis and reaction diagnosis**

An **example** of complete monitoring (interlock diagnosis, action diagnosis and reaction diagnosis) is illustrated in Fig. 1-6.



Process requirement — **Interlock diagnosis**

Trigger:	Manual
	(A) Pushbutton = P
Trigger time:	1 * 100ms
Action:	Blower motor = 1
	(A) Emer. off = 0
	(A) Safety screen = 1

Process function — **Action diagnosis**

Action:	Blower motor = 1
	(A) Emer. off = 0
	(A) Safety screen = 1
Monitoring time:	50 * 100ms
Reaction:	Pressure sensor = 1

Final process status — **Reaction diagnosis**

Reaction:	Pressure sensor = 1
Tolerance time:	2 * 100ms
Stop condition:	Tank pressure = 1

Fig. 1-6 Example: blower monitoring

In this example, the interlocks are also to be monitored, they must therefore be specified with the action.

The monitoring is completed when

- the process element has been run through, or
- the monitoring is stopped, e.g. the stop condition becomes active.

Note on **release:**

- If a release condition is not fulfilled, the monitoring of the corresponding process element is not started.

  Example:
  The power supply for an expansion unit is used as the release for a subprocess.

- If the release changes from "present" to "not present" during the monitoring of a process element, this has the following significance for the process element:

  a currently active timer is cleared
  an error is signalled as cleared,
  the monitoring is stopped,
  all edges are cleared.

Monitoring starts again from the beginning only when the release is valid again.

Monitoring process requirements with interlock diagnosis:

The **process requirement** includes the following:

- trigger,
- trigger time and
- action.

The **action** consists of the logical combination of several signals. A distinction is made within these signals between the **trigger** and **interlocks.** The trigger actually initiates the action, the interlocks serve as the required process environment.

If all the terms of the trigger are fulfilled, the action must be activated during the trigger time. The trigger time t can also be 0, i.e. the action starts in the same cycle as the trigger. If t is greater than 0, the action does not start immediately. If the action does not start within the trigger time, an error is signalled.

"Interlock error with t = 0"

or

"Interlock error with t > 0."

Example (see Fig. 1-3):

If manual operation is switched on and the manual switch is activated, the action must start within the trigger time (t = 0 or t > 0). The action can only be activated if the stipulated interlocks are fulfilled. If the emergency stop switch has been pressed, or if the safety screen is not mounted, an interlock error is signalled.

Trigger monitoring

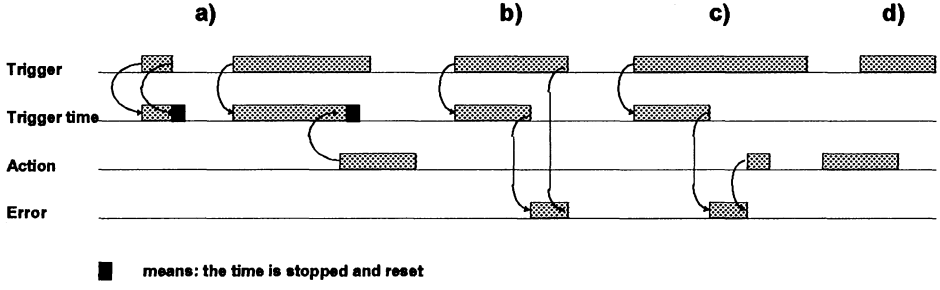


Fig. 1-7 Monitoring triggers in interlock diagnosis

a) When the trigger is activated, the trigger time is started. The trigger time is stopped when the trigger is no longer active or the action starts.

b) When the trigger time has elapsed, without the action starting, an error will be signalled as long as the trigger is active.

c) When the trigger time elapses, an error is signalled until the action starts.

d) When the action starts, before the trigger becomes active, no error is signalled.

Monitoring process functions with action diagnosis:

A **process function** consists of the following:

- action,
- monitoring time and
- reaction.

When the **action** becomes active, the monitoring time is started. The monitoring time is stopped when the **reaction** becomes active. If an error occurs, the action diagnosis signals "reaction not reached."

There are two types of process function monitoring, as follows:

- **motive:** the monitoring continues as long as the action is active and the stop condition has not been fulfilled.

  Example of motive process function (Fig. 1-1):
  As long as the output is set, the valve remains open and the tank continues to fill.

- **pulse-dependent**: the action is monitored from its start until the reaction or stop condition is activated (even if the action disappears in the meantime).

  Example of pulse-dependent process function:
  A pulse at the output of the PLC controls a mechanically latching relay. Because of the latching, the valve remains open, even when the signal disappears at the output.

- **motive/pulse-dependent**: the monitoring is motive until the first term of the reaction becomes active. Following this, it is pulse-dependent until the reaction is activated or the stop condition is fulfilled.

  Example of motive/pulse-dependent activity:
  Carriage: the carriage motor is switched off at limit switch 2. The final position (limit switch 3) is reached by the carriage's kinetic energy. While this process function is being monitored, it therefore changes from motive to pulse-dependent.

  Action: motor on
  Reaction: limit switch 2 = P AND limit switch 3 = P

Motive process function monitoring

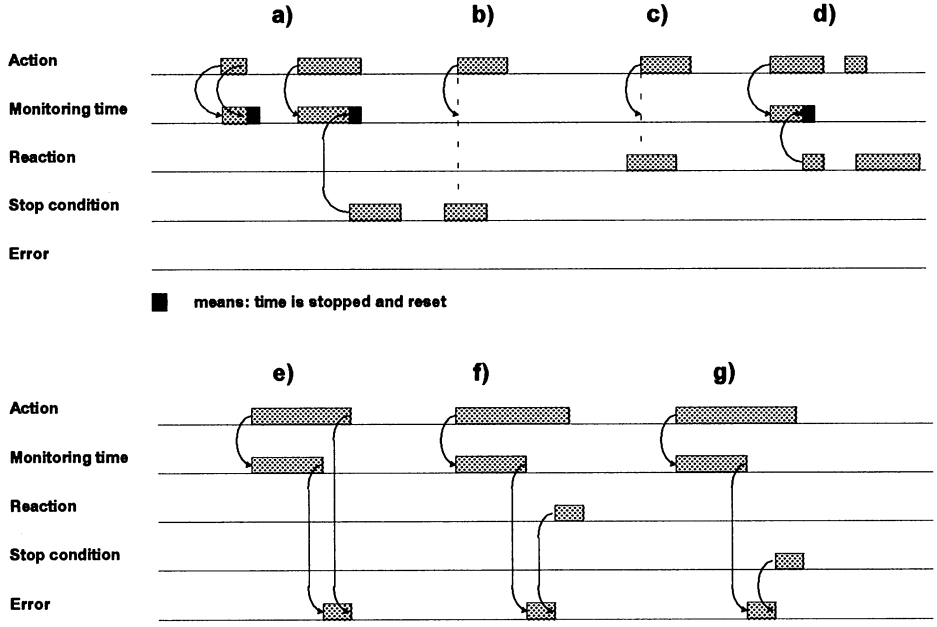


Fig. 1-8 Motive process function monitoring in action diagnosis

a) When the action begins, the monitoring time is started. It is stopped when the action stops or the stop condition is fulfilled.

b) The action does not trigger a monitoring time if the stop condition is fulfilled.

c) The action does not trigger a monitoring time if the reaction is active.

d) When the reaction becomes active, the monitoring time is stopped. The monitoring of the final status is started; the action is no longer monitored.

e) When the monitoring time has elapsed without the reaction being activated an error is signalled.

f) If the reaction is activated at a later point in time, the current error is cleared.

g) If the stop condition is fulfilled at a later point in time, the current error is cleared.

Pulse-dependent process function monitoring

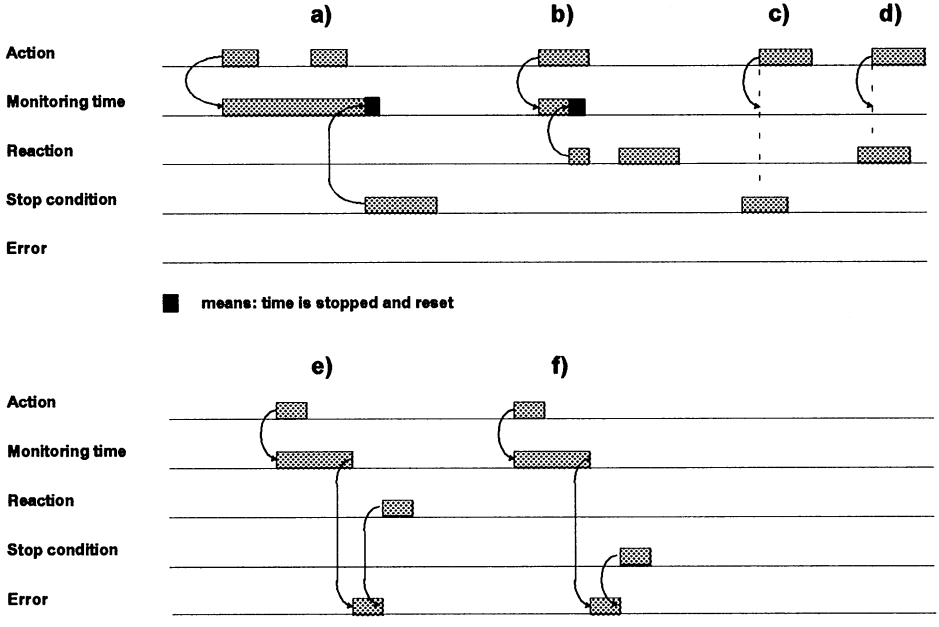


Fig. 1-9 Pulse-dependent process function monitoring in action diagnosis

a) The monitoring time is started by the signal edge of the action. It is completed when the stop condition is fulfilled.

b) The monitoring time is stopped when the reaction becomes active.

c) If the stop condition is active, the monitoring time cannot be started.

d) If the reaction is active, the monitoring time cannot be started.

e) When the monitoring time has elapsed, an error is signalled until the reaction becomes active.

f) When the monitoring time has elapsed, an error is active until the stop condition is fulfilled.

Motive/pulse-dependent process function monitoring

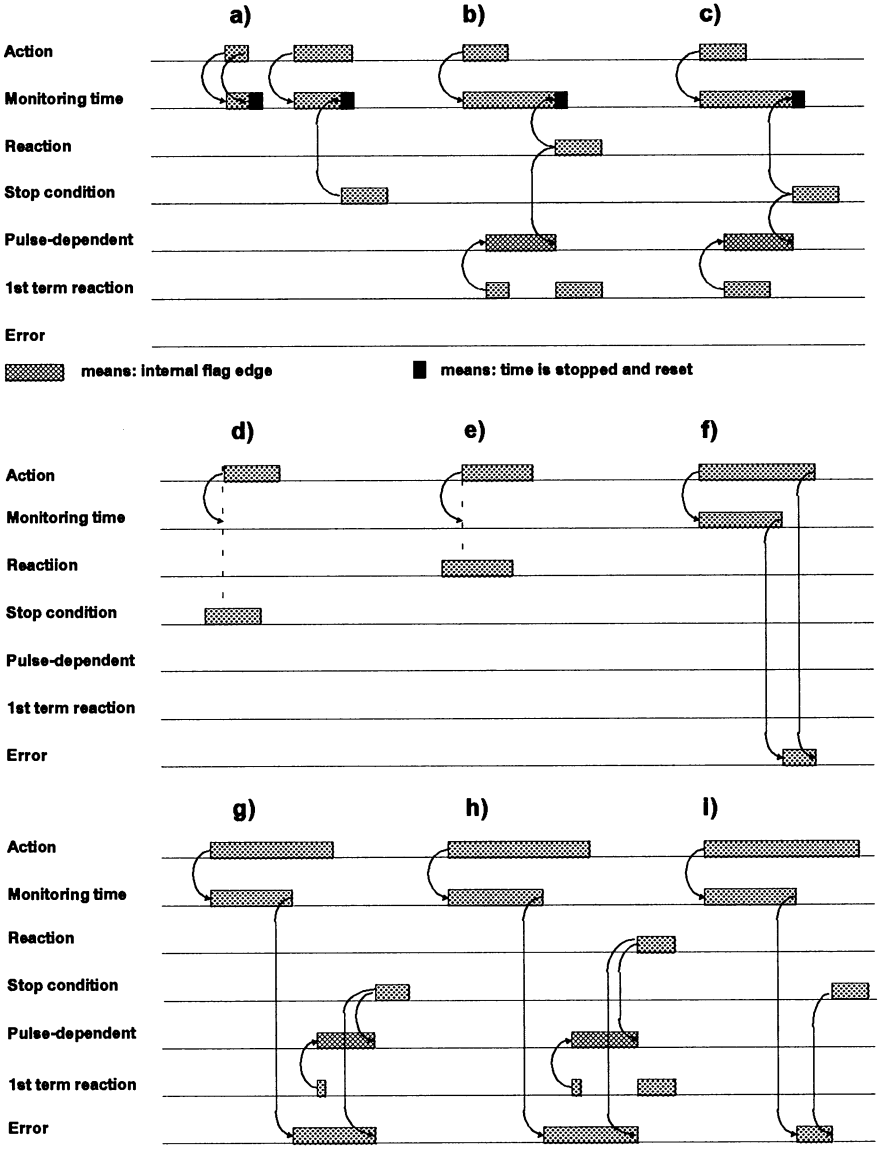


Fig. 1-10 Motive/pulse-dependent process function monitoring in action diagnosis

"1st term reaction" means: this is the first term in the order in which the terms become active, not the order in which they are programmed in the setpoint data.

a) Only motive monitoring, since the first term of the reaction does not become active. The monitoring time is stopped when the action is completed or when the stop condition is fulfilled.

b) The monitoring becomes pulse-dependent when the first term of the reaction becomes active, until the reaction is complete.

c) The monitoring becomes pulse-dependent when the first term of the reaction becomes active. The monitoring is completed when the stop condition is fulfilled.

d) The monitoring time does not start, since the stop condition is fulfilled before the action becomes active.

e) The monitoring time does not start, since the reaction is present before the action becomes active.

f) If the monitoring time elapses without the reaction being active, an error is signalled until the action is completed.

g) If the monitoring time elapses without the reaction being active, an error is signalled. When the first term of the reaction becomes active, the monitoring becomes pulse-dependent. The stop condition stops the monitoring and clears the error.

h) If the monitoring time elapses without the reaction being active, an error is signalled. When the first term of the reaction becomes active, the monitoring becomes pulse-dependent. The reaction terminates this status and clears the error.

i) When the monitoring time elapses without the reaction being active, an error is signalled until the stop condition is fulfilled.

Monitoring final process statuses with reaction diagnosis:

The **final process status** consists of the following:

- reaction
- tolerance time
- stop condition

The monitoring of the final process status (example see Fig. 1 - 6, Blower monitoring) requires that a process function has already been completed. The reaction must remain active until the stop condition is fulfilled. The stop condition stops the monitoring of the final process status. If the reaction changes before the stop condition is fulfilled, an error is signalled. The reaction diagnosis signals "final status exited illegally." The reaction can become inactive temporarily provided it does not exceed the tolerance time.

You can also use the reaction as a start condition. After each cold restart on the CP 552, the start condition is checked by the CP 552 (provided that the setpoint data element is released and the start condition has been selected). If the start condition is not fulfilled, an error is signalled.

Example:

The equipment can only be started up when all the moving parts are in their initial position.

Final status monitoring

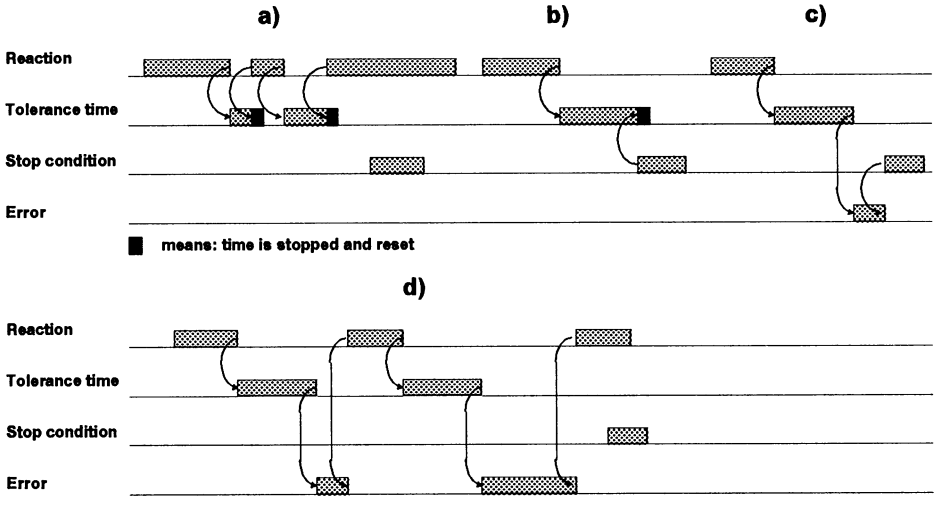


Fig. 1-11 Final status monitoring in reaction diagnosis

a) If the reaction becomes briefly inactive without exceeding the tolerance time, no error is signalled. The stop condition stops the monitoring.

b) If the reaction becomes inactive, the tolerance time starts and is stopped by the stop condition.

c) If the reaction is inactive and the tolerance time has elapsed, an error is signalled until the stop condition is fulfilled.

d) When the reaction is not active and the tolerance time ends, an error is signalled. If the reaction becomes active again, the error is cleared. The stop condition stops the monitoring.

1.2.2 Monitoring Statuses

Fig. 1-5 in Section 1.2 "monitoring pairs of limit switches" is an **example** of status monitoring.

The statuses are defined by the logical ANDing and ORing of terms. When monitoring statuses, you can specify a tolerance time, i.e. the undesired status can exist for a certain time. If it exists for longer than the tolerance time t, the error "illegal status t = 0" or "illegal status t >0" is signalled.

 **Note:**

When monitoring **statuses,** you define the **illegal** statuses. When monitoring **dynamic functions** you describe the **error-free** process functions.

Notes on **release**

- The monitoring of the status is not started unless the release condition is fulfilled.

  Example:
  The power supply for an expansion unit is used as the release for the corresponding subprocess.

- If the release is active, a check is made to establish whether the illegal status exists. If it does, the tolerance time is started.

- If the release changes from "present" to "not present" during the monitoring of a status, the results are as follows:
  - a currently active tolerance time is cleared,
  - any error is signalled as cleared,
  - the monitoring is stopped.

Monitoring Statuses

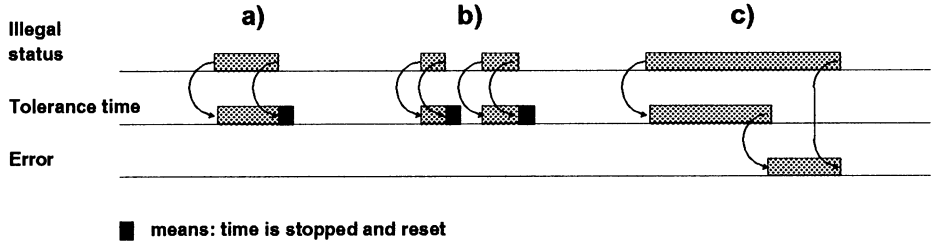


Fig. 1-12 Monitoring statuses

a) The illegal status exists for a shorter time than the tolerance time: no error.

b) The tolerance time is started when the illegal status appears and is stopped when the status ceases to exist.

c) If the illegal status exists for longer than the tolerance time, an error is signalled. When this status no longer exists, the error message is also cleared.

1.3 Generating Setpoint Data Elements

The setpoint data elements, which represent the model of the process to be monitored, are generated on the programmer in the LAD, CSF, STL or COM 552 package. In the COM 552 programming package you can also add setpoint data elements to already existing STEP® 5 user programs. More detailed information about this topic can be found in this manual in the User's Guides "LAD, CSF, STL Package with Process Error Diagnosis" and "COM 552 Programming Package".

The setpoint data elements are identified by the block type, block number and segment number. This makes the following tasks easier:

- generation of setpoint data elements
- debugging
- documentation of your STEP® 5 user programs and the corresponding setpoint data elements.

1.4 Error Identifier

When you generate setpoint data elements, you can assign an error identification number to each setpoint data element. This number has four digits **(maximum 4095).** If the CP 552 recognizes a process error, the CPU fetches the error identification number and additional information about the error type and status of the error message (see RECEIVE DIRECT 200). The STEP® 5 user program can now interpret the error identifier. For further information, refer to Sections 2.2.2 and 2.3.2.

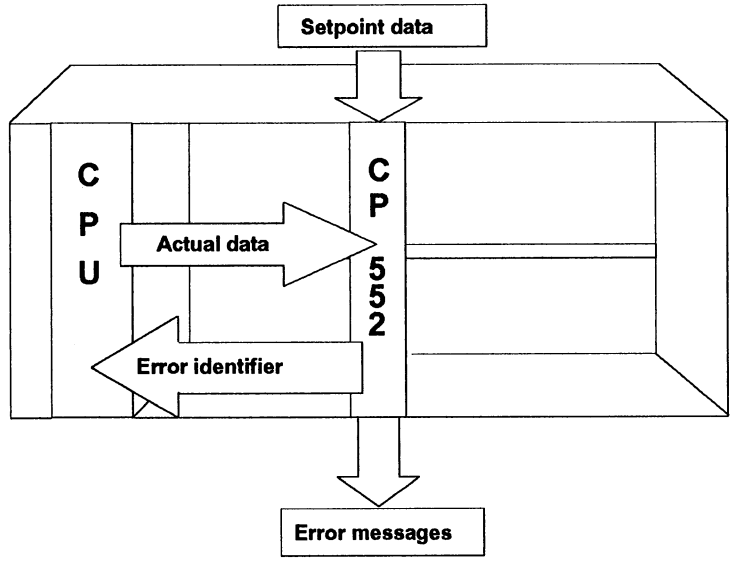


Fig. 1-13 Error identifier

1.5 Error Display on the PG

1.5.1 Types of Messages

The CP 552 handles the following types of message:

- **Process error messages**

  Process errors are detected by the CP 552 by comparing the setpoint and actual data and are then displayed on the display unit. Examples of error displays can be found in Chapter 3 "Appendix - Examples of Error Display on the PG."

- **System messages**

  These are messages about statuses or errors in a CP 552, and are generated by the **CP 552** and sent to the display unit (example: "CP 552 STOPPED").

- **Process control messages**

  These are messages from the **CPU** of the PLC, which are sent to the CP 552. The CP 552 passes these messages to the display unit. Notes on the generation of process control messages can be found in this manual in the User's Guide "Displaying Process Control Messages."

The maximum number of messages which can be stored in the CP 552 and PG depends on the length of the individual messages. This length is increased by the following factors:

- the number of operands signalled as incorrect and
- the length of the user comments belonging to the setpoint data elements.

In the CP 552, a maximum of **27** messages can be stored and in the PG a maximum of **40**.

The messages can be transferred from the CP 552 to the PG in either of the following ways:

- **locally** via the AS 511 interface or
- **centrally** via a CP 535 and the SINEC H1 bus.

1.5.2 Single Diagnosis

For single diagnosis, you require **one** PG as a display unit **per** CP 552. The PG polls the CP 552 for messages.

The data can be transferred in either of the following ways:

- **locally** via the AS 511 interface (Figs. 1-14 and 1-15) or
- **centrally** via the AS 511 interface of the CP 552, the PG channel of the CP 535 and the SINEC H1 bus (Figs. 1-16 and 1-17).

The following messages can be displayed:

- process error messages
- system messages
- process control messages.

Single diagnosis locally via the AS 511 interface

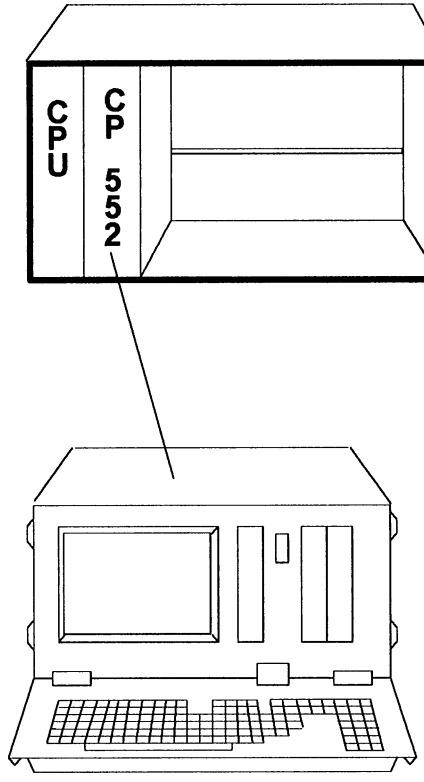


Fig. 1-14 Single diagnosis locally via the AS 511 interface;
PG connected directly to the CP 552

Single diagnosis locally via the AS 511 interface

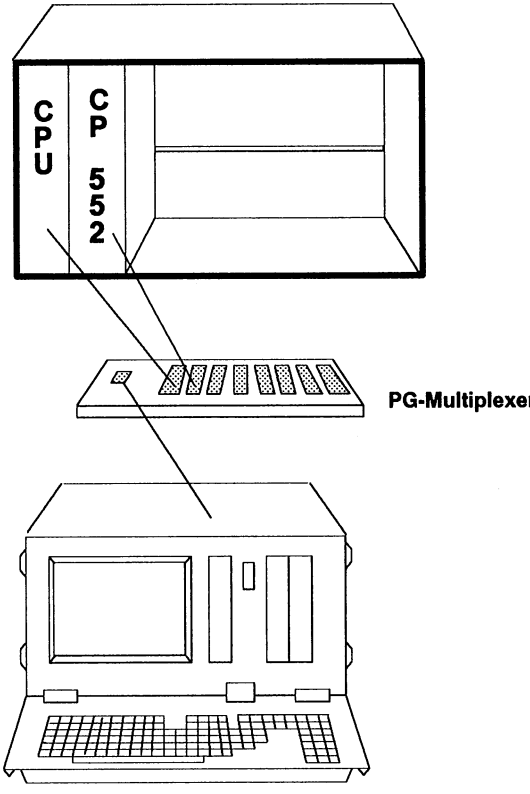


Fig. 1-15 Single diagnosis locally via the AS 511 interface;
PG connected via multiplexer (or KOR C coordinator with S5-135U) with the CPU and CP 552.

Single diagnosis on SINEC H1 via the PG channel of the CP 535

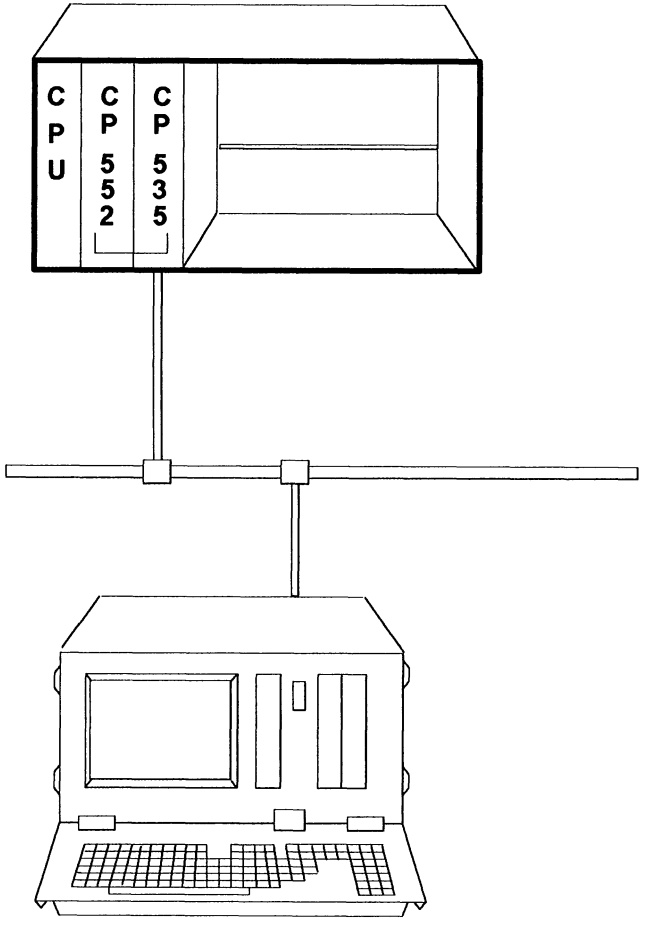


Fig. 1-16 Single diagnosis on the SINEC H1 bus

Single diagnosis on SINEC H1 via the PG channel of the CP 535

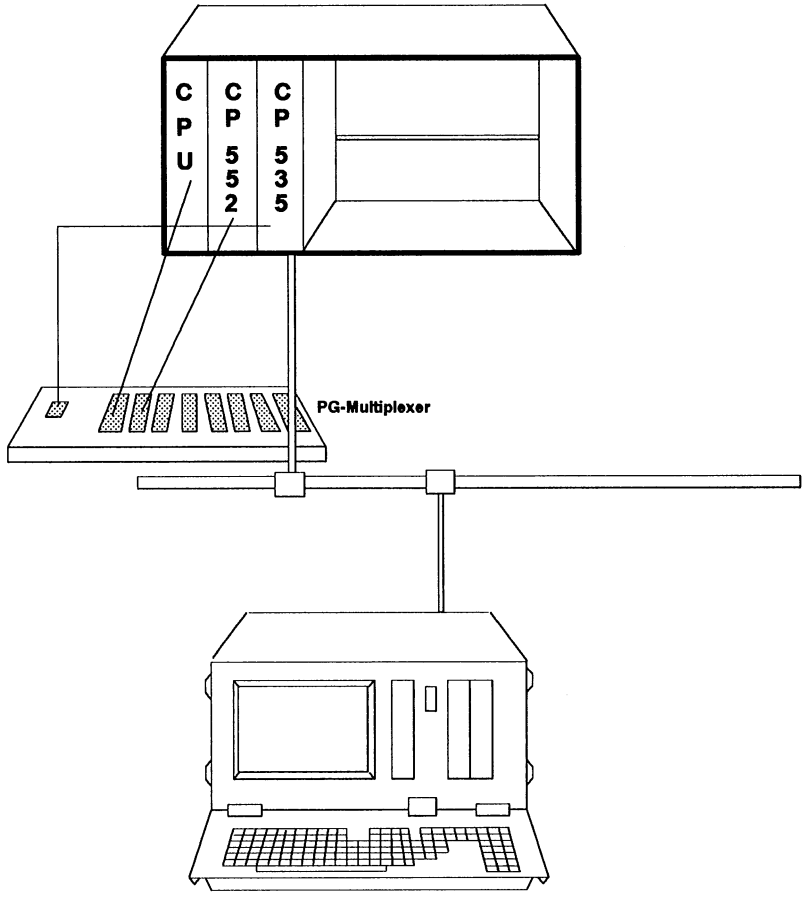


Fig. 1-17 Single diagnosis on the SINEC H1 bus
PG connected via multiplexer (or KOR C coordinator with S5-135U) with CPU and CP 552

If you use the **PG multiplexer** (Figs. 1-15 and Fig. 1-17) or **KOR C** with the S5-135U, you have the following advantages:

- If an interlock error occurs, you can display the incorrect segment in the STEP® 5 user program via the segment status. In this way, you can check flag assignments as far back as the inputs.

  For further information, refer to your programmer manual.

- If there is a system error (CPU stopped), you can display the BSTACK/ISTACK of the CPU directly.

1.5.3 Group Diagnosis

Group diagnosis is only possible via the SINEC H1 bus. With group diagnosis, you can display the following:

- Process error messages
- System messages
- Process control messages
- Segment status (only with the PG multiplexer)
- BSTACK/ISTACK (only with PG multiplexer)

Standard function blocks on the CPU handle the distribution of messages to the connected display units. Information on ordering function blocks for process error diagnosis with the CP 552 diagnostic processor can be found in the ordering data.

Group diagnosis on SINEC H1

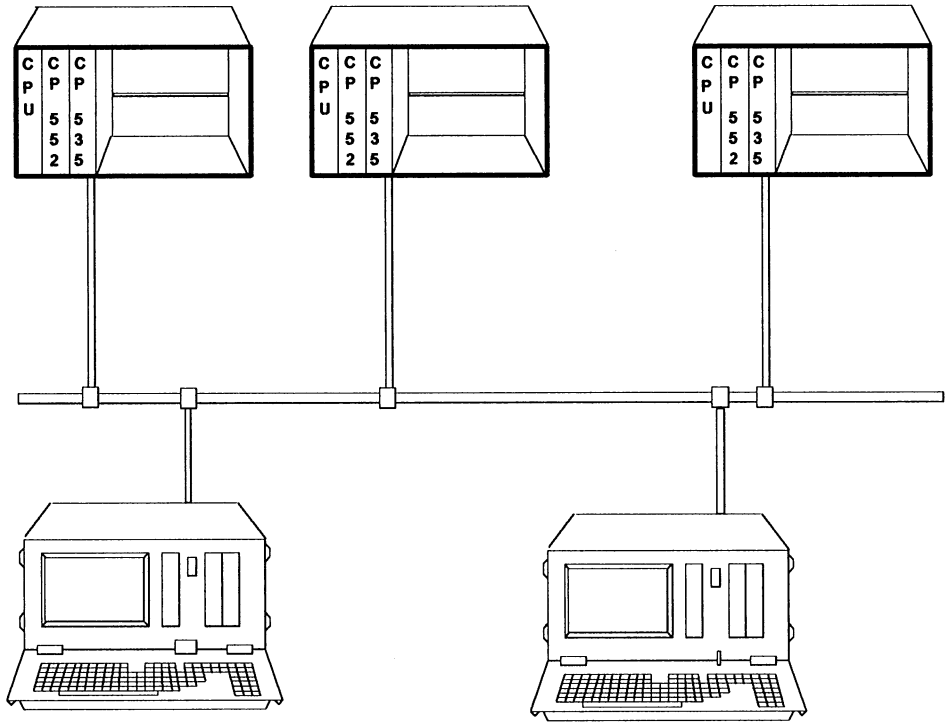


Fig. 1-18 Group diagnosis on the SINEC H1 bus

Features of the group diagnosis:

- Up to 16 diagnostic processors can display their messages **on one programmer** -- known as group display.

- Each diagnostic processor can display its messages **on up to 8 different programmers**. The CP 552 can even display the same message on several programmers.
  The CP 552 transfers the messages of the CPU FB SEND, to which you have assigned parameters, handles the distribution of the individual messages to the appropriate display units.

  You stipulate the display units for process error messages in the LAD, CSF, STL or COM 552 package in the "display locations" field. To display the process control and system messages, you must assign parameters accordingly for the CP 552 in the COM 552 programming package.
  Example of display unit assignment:
  Device numbers: 8 7 6 5 4 3 2 1
  Display locations: 0 1 0 0 1 0 0 0
  In this case, devices 4 and 7 have been selected as the display locations.

  The assignment of device numbers to display units is not fixed, but applies rather from the point of view of each individual CP 552. The number of display units in one system is not limited.

- If there is no error, the PG does not poll, i.e. does not increase traffic on the SINEC H1 bus.

- The PG channel of the CP 535 remains free for other applications, e.g. for the status display of the STEP® 5 user program on another PG.

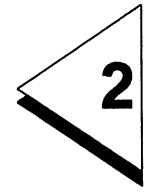
1.6 Error Display on the Local Monitor

You can connect a monitor to the VIDEO interface of the CP 552. This monitor can then display the following messages

- process error messages
- system messages
- process control messages.

For further information (particularly regarding installation) refer to this User's Guide "Error Display on the Local Monitor."

Installation



Contents

Once you have prepared your module for use as described in the operating instructions "CP 552 Diagnostic Processor" and have inserted it in the PLC frame, you can install the software for the CP 552. This procedure is described in the following sections.

2.1 Generating Setpoint Data Elements

The first step is to generate the setpoint data elements on the programmer. These must correspond to the process and the required monitoring functions. If you would like to generate the setpoint data elements at the same time as the STEP® 5 user program, you should perform this directly in the LAD, CSF, STL package (see User's Guide "LAD, CSF, STL Package with Process Error Diagnosis"). Using the programming package COM 552 you can generate the setpoint data separately from the STEP® 5 user program (see User's Guide "COM 552 Programming Package").

2.2 Local Error Display on the PG via the AS 511 Interface

Only **single diagnosis** is possible using the AS 511 interface. The hardware configuration required is described in Section 1.5.2 (Figs. 1-14 and 1-15). Fig. 2-1 shows the programming packages required for single diagnosis via the AS 511 interface.

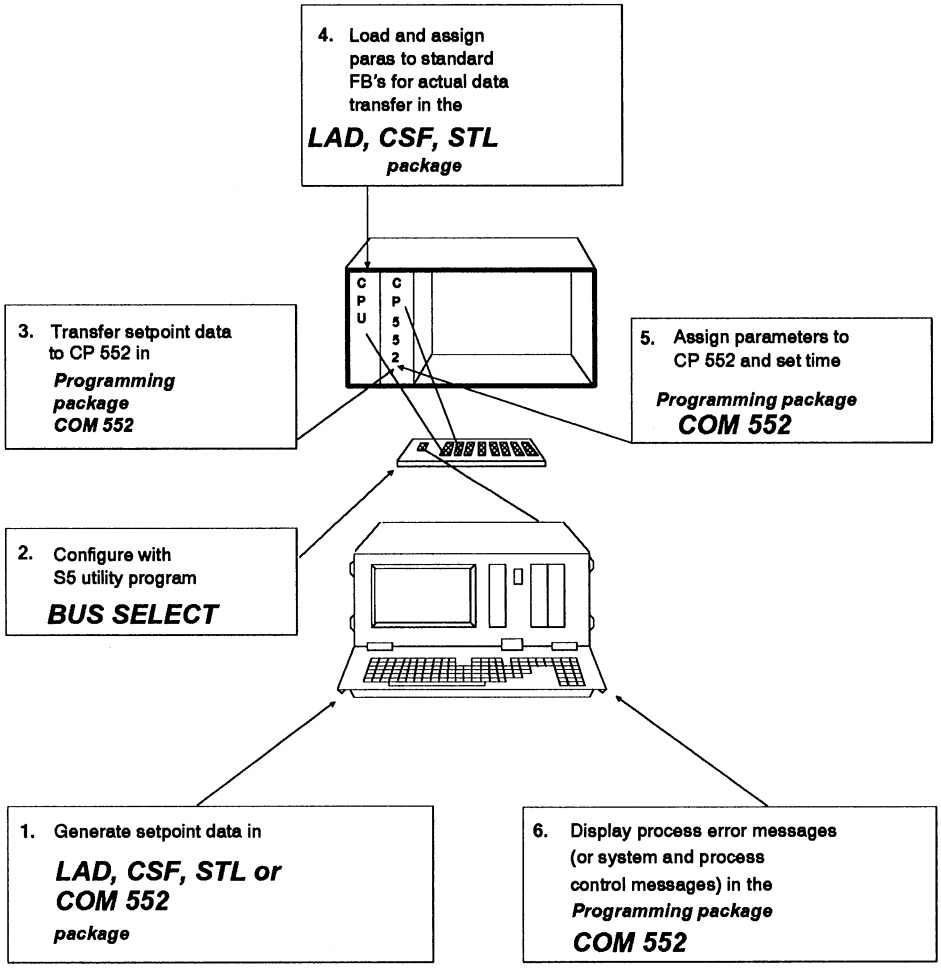


Fig. 2-1 Local single diagnosis via the AS 511 interface

2.2.1 Blocks on the CPU

For the communication between the CPU and CP 552 you must transfer several function blocks and data blocks to the CPU in the LAD, CSF, STL package (see ordering data). These blocks are as follows:

FB ANLAUF	Start-up block for FB KOMCP552
FB KOMCP552	Communication block CPU / CP 552 for actual data transfer
HDB SYNCHRON	Handling block *)
HDB SEND	Handling block *)
HDB RECEIVE	Handling block *)
HDB CONTROL	Handling block *)
	*) these handling blocks are called in the FBs ANLAUF and KOMCP552.
DB PARAM	Internal data for the FBs
DB BLOCK	Blocking for actual data transfer
DB E/A	Data of the inserted inputs/outputs (digital)
Only with S5-135U and S5-155U: **DB SAMMEL**	Internal collection of data for transfer to CP 552.

FB KOMCP552 must be called **last in OB 1** to ensure that the actual data (inputs, outputs and flags) are transferred to the CP 552 at the end of the PLC cycle.

FB ANLAUF must be called in the organization blocks **OB 20/21/22**. A parameter used when calling FB ANLAUF is the interface number of the CP 552. If you are using several CPs or IPs in your programmable controller, these must be assigned different interface numbers. For further information regarding the interface number, refer to section 3.5 in the instructions "CP 552 Diagnostic Processor". More information about the blocks can be found in Section 2.6 "Calling and Assigning Parameters to Blocks on the CPU."

2.2.2 Error Identifier

Structure of the error identifier

Bit 15	**Status of the error message**
1	error present
0	error cleared

Bit 14	**Bit 13**	**Error type**
0	0	illegal status
0	1	interlock error
1	0	reaction not reached
1	1	final status exited illegally

Bit 12	**Reserved**
Bit 11 to bit 0	**Error identifier number**

Error identifier numbers in the setpoint data element:

The numbers 0 to 4095 mean that an entry is possible.
Here 0 ⇒ means no error identifier present
and 1 to 4095 ⇒ error identifier

Please note that an error identifier may only be used **once**. If several setpoint data elements have the same error identifier, the status bits cannot be evaluated because an error identifier cannot be matched to a message.

If the error identifiers of the CP 552 are to be evaluated by the CPU, you must also link the

RECEIVE DIRECT with **job number 200**

in your STEP® 5 user program. With this RECEIVE DIRECT 200, the CPU receives the error identifiers entered since the last RECEIVE DIRECT 200 as soon as an error with an error identifier exists, as follows:

DW0: error identifier 1
DW1: error identifier 2
DW2: error identifier 3

A "0" in the data word signifies that there is no error identifier present.

If more than three error identifiers have occurred since the last call, they are stored on the CP 552 (up to 27 error identifiers) and transferred with the next RECEIVE jobs.

Note:

The error identifiers must be evaluated **immediately** following the transfer, since the "old" identifiers are overwritten by the next RECEIVE DIRECT 200.

2.2.3 Assigning Parameters to the CP 552

The **SYSID** (system identification area) of the CP 552 is assigned parameters in the programming package **COM 552.**

If process error diagnosis is only being used **locally** (not via SINEC H1) all the **display locations** must be set to **0** in the SYSID of the CP 552.

If you wish to display the BSTACK and ISTACK when a segment status process error occurs, you must enter a **plant identifier** in the SYSID. The plant identifier is used to make an automatic path selection to establish the connection PG/CPU (see Section 2.2.4 Path File).

The SYSID of the CP 552 also contains the firmware release of the CP 552.

2.2.4 Path File

Fig. 2-2 shows the configuration with which you can display not only error messages, but also the segment status, BSTACK and ISTACK via the AS 511 interface and the multiplexer (or KOR C with the S5-135U).

To select the path to the CPU and to the CP 552 automatically, you require the **utility BUS SELECTION** (refer also to your programmer manual). This program is called in the S5 screen form **SELECT PACKAGE** in which a path file with a path to the CP 552 and a path to the CPU is created. The path name to the CPU must be entered in the SYSID of the CP 552 as plant identifier (see Section 2.2.3). The path name to the CP 552 can be selected freely.

If you would like to display

- segment titles,
- statement comments and
- segment comments

with the segment status, you must enter the name of the program file belonging to the CPU using the supplementary function when editing the path file.

Path file - to display the segment status, BSTACK and ISTACK via multiplexer

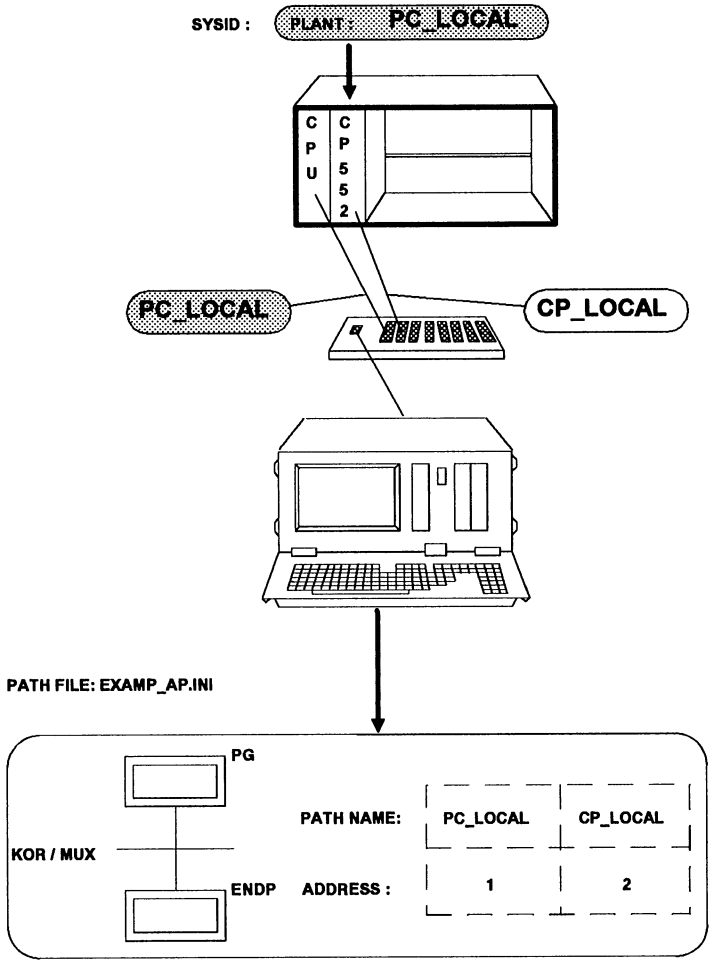


Fig. 2-2 Path file

2.2.5 Transferring Setpoint Data Elements and Displaying Process Errors

The generated setpoint data elements are transferred to the CP 552 in the programming package COM 552. The CP must be in the STOP mode. Following the transfer, switch it to RUN. If you are unable to switch the CP 552 to RUN, you can have possible causes of error displayed using the information function "CP-INFO" in the SCREEN FORM EDITOR.

Now exit the SCREEN FORM EDITOR. Select "SINGLE" as the CONFIGURATION presetting. Change to the function CURRENT MESSAGES. If the CP 552 is set to RUN, you can now display the process error messages. For further information about error displays, refer to the User's Guide "COM 552 Programming Package".

2.2.6 Setting and Reading the Time/Date

There are two ways of setting and reading the time and date on the CP 552:

Method 1: using the DATE/TIME function in **COM 552** you can read and set the current values for the time and date.

Method 2: using the **handling blocks**, the clock of the CP 552 can be set or read by several CPUs. The CP 552 makes the following jobs available to the CPUs:

- **Set time/date: SEND DIRECT** with **job number 218**
- **Read time/date: RECEIVE DIRECT** with **job number 218**

The clock in the CP 552 can be used as **time master** for the whole system or as **time slave** within the system.

The data transferred for setting and reading the time/date is structured as follows:

DW n:	| Master identifier	| 0
DW n + 1:	| 1/10 second	| seconds
DW n + 2:	| minute	| hours
DW n + 3:	| day	| month
DW n + 4:	| year	| 0

The values for the time and date are entered in BCD code.

The master identifier "1" means that the CP 552 is master for the time/date. This master identifier has no significance for the CP 552 itself, it is only evaluated by the CPUs and other CPs.

Setting the time/date: SEND DIRECT 218

The system time (time/date) is sent to the CP 552 with SEND DIRECT 218. The data transferred is contained in a data block and structured as shown above. This function is executed in one run-through of the SEND DIRECT handling block. The syntax of the values for the date and time in the data transferred is not checked.

The parameter QLAE can be "= 1" or "= 5". If **"QLAE = 1"** only the master identifier is transferred. If **"QLAE = 5"** the time/date are also transferred. The parameter QLAE can only have these two values. Parameters are assigned to SEND DIRECT 218 as follows:

NAME:	SEND
SSNR:	interface number of the CP 552
A-NR:	job number 218
ANZW:	address of the condition codeword
QTYP:	source data type in the CPU (DB, DX)
DBNR:	number of the data block, in which the time/date are stored
QANF:	start address of the data in the data block
QLAE:	length of the data to be transferred in words (1 or 5 words)
PAFE:	error condition code for parameter errors.

Reading the time/date: RECEIVE DIRECT 218

The time/date of the CP 552 are read using RECEIVE DIRECT 218 (refer to SEND DIRECT 218). The RECEIVE DIRECT 218 has parameters assigned as follows:

NAME:	RECEIVE
SSNR:	interface number of the CP 552
A-NR:	job number 218
ANZW:	address of the condition codeword
QTYP:	source data type in the CPU (DB, DX)
DBNR:	number of the data block in which the date and time are stored
QANF:	start address of data in the data block
QLAE:	length of the transferred data in words (1 or 5 words)
PAFE:	error display for parameter error

2.3 Displaying Errors on the PG centrally via SINEC H1

Via the SINEC H1 bus, both group diagnosis and single diagnosis are possible. For error displays via the SINEC H1 bus, you require a **bus interface module CP 535 in each PLC** and **a bus interface module CP 536 in each PG**.

The possible hardware configurations are described in Figs. 1-16, 1-17 (single diagnosis) and 1-18 (group diagnosis). Fig. 2-3 shows the programming packages required for error display via the SINEC H1 bus.

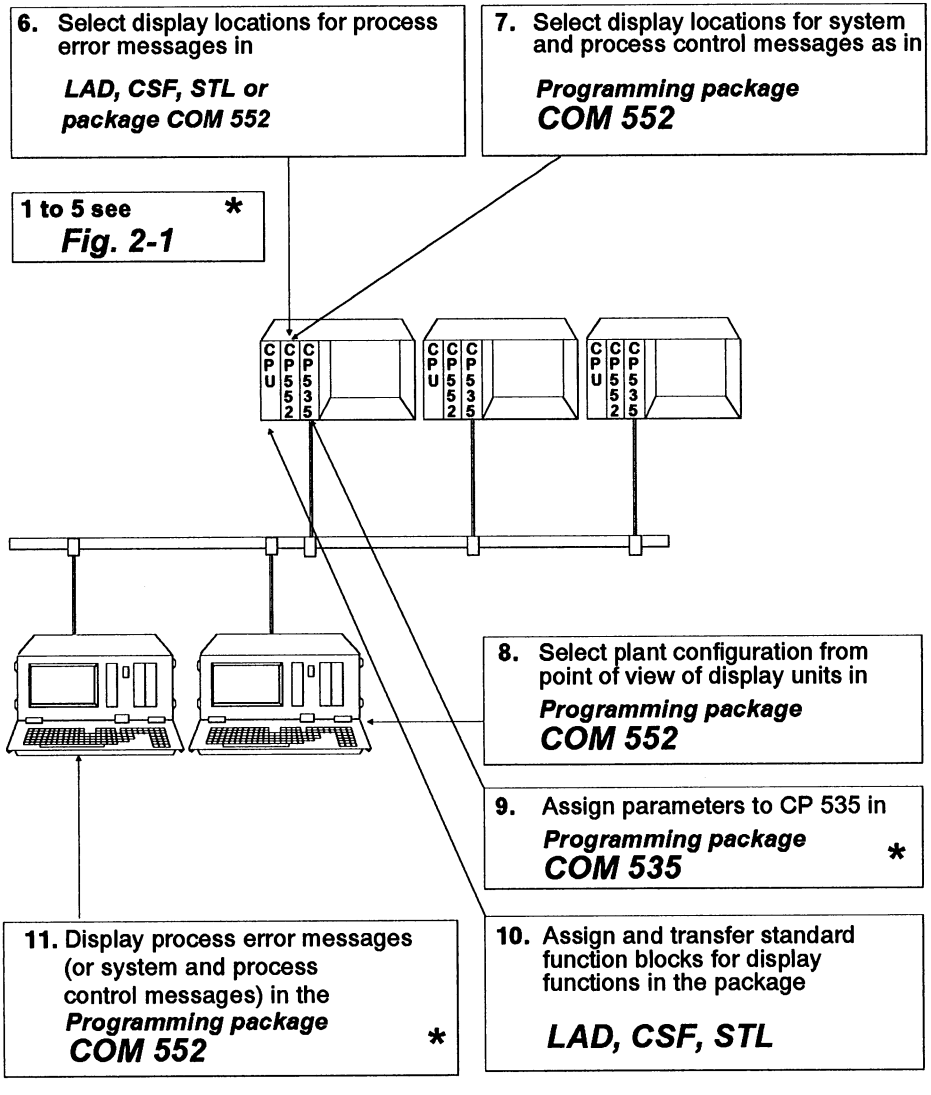


Fig. 2-3 Group diagnosis on the SINEC H1 bus (1 - 11)
* Single diagnosis on the SINEC H1 bus (1 to 5 or 9 and 11)

2.3.1 Project Planning Overview

Group diagnosis enables you to seletively send messages to display units. For example, let us assume that a message A is to be displayed on display unit PG1, a message B on display unit PG2, system messages on PG1 **and** PG2 and process messages on PG3. To determine the path followed by a message from CP 552 to the desired display unit, you will have to configure the modules involved in the communication; a summary of their interactions and interdependencies is given below.

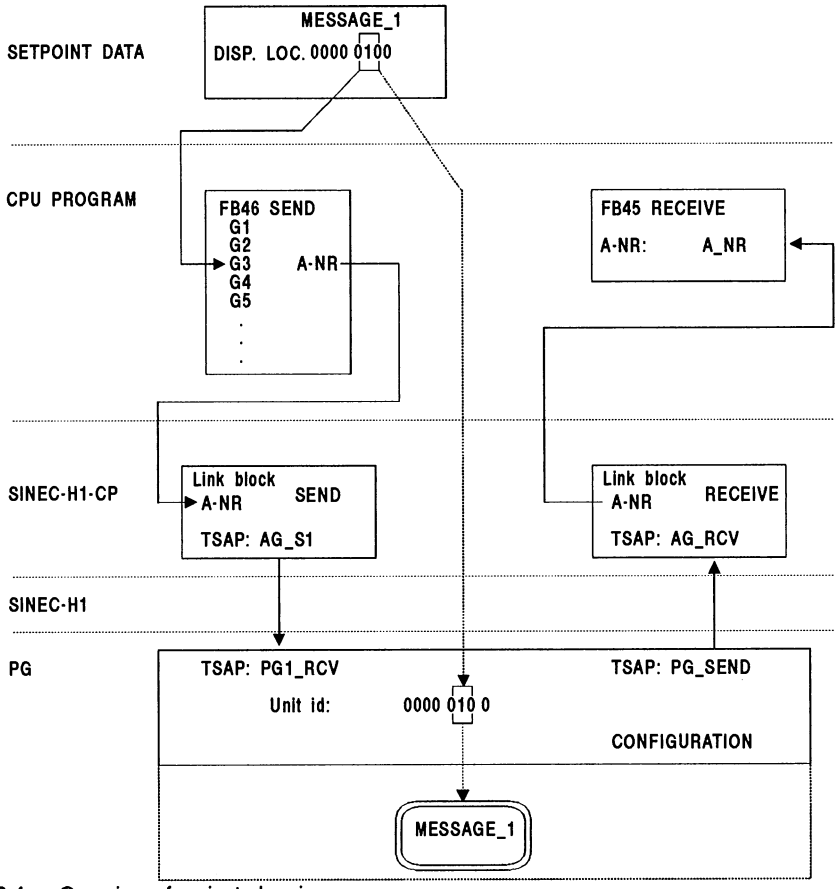


Fig. 2-4 Overview of project planning

Fig. 2-4 depicts, in the group diagnosis mode, the mechanism and the modules involved in the communication between the diagnostic processor and display unit.

In the DISPLAY LOCATIONS field of the setpoint data element, the bit position corresponding to the DISPLAY UNIT at which the **process error message** is to be displayed, is changed to "1". The message can be displayed on several display units by changing several positions to "1".
The DISPLAY LOCATIONS for **system and process control messages** are selected according to the same pattern in the SYSID screen form.
The bit position of the DISPLAY LOCATION field correlates with the unit number in the parameter block of FB 46 "SEND", i.e. a setpoint data element featuring a "1" at bit position 2 of the DISPLAY LOCATION field is transferred via the JOB NUMBER of unit 2 on calling FB 46.

A link block is programmed to a display unit under this JOB NUMBER in the SINEC-H1-CP (CP 535, CP 143). This effects the actual transfer of the message via the SINEC-H1 bus.

The addressing mechanism of the SINEC-H1 bus is characterized by the unit address of the bus partner (Ethernet addresses) and by the specifications which uniquely define the link programmed in the respective partner (TSAP = Transport-Service-Access-Point). Apart from your own TSAP, a link also features the TSAP of the other partner, these must be known to both partners:

Unit	TSAP own	<--\	/-->	TSAP own	Unit
A	TSAP remote	<--\	/-->	TSAP remote	B

with the DISPLAY LOCATION featured in the message received.

If the identifiers correlate at the position defining the display unit, the message will be displayed on the display unit.

UNIT ID	0001 ID of the display unit 0000
DISPLAY LOCATION	0011 Setpoint data element 0000
(the message is displayed by means of AND logic combinations of both identifiers, if the result is not equal to 0!)	0001 The message is displayed 0000 on the display unit

Data traffic with the reverse transmission direction from display unit to the diagnostic processor is effected with the "unspecified Ethernet address" 00.00.00.00.00 H.
All RECEIVE link blocks for the group diagnosis in PLCs in a network hence operate with the "remote" Ethernet address 00.00.00.00.00 H and the same remote TSAP (in the example: PG_SEND).
Accordingly, all PG links must feature a similar TSAP: LTSAP(s) (e.g.: PG_SEND).
This link, called a BROADCAST, enables the display units to simultaneously transmit jobs to all diagnostic processors. One example of this is the "Total error request" which the PG sends when it starts to operate as a display unit in order to request from all diagnostic processors the waiting messages destined for the PG.

 **Note**:

In the case of multiprocessor operation, in particular if two or several CPUs are working together with one CP 535, a different LTSAP must be available for each page of the CP 535 used.

The two LINK BLOCKS defining a link for SEND and RECEIVE in the SINEC-H1 CP have a corresponding CONFIGURATION in the PG configuration file.

To display the segment status, the ISTACK and the BSTACK of the CPU from the COM 552, a path file must be created, containing paths to the diagnostic processor **and** the CPU. The PATH NAME of the CPU path must be identical to the plant name selected in the CP 552-SYSID and to the plant name of the configuration (in the configuration file). If one of previously mentioned functions is triggered, the COM breaks the link to the diagnostic processor and re-establishes the link on the basis of the plant name specified in the CONFIGURATION, which also corresponds to the path name to the CPU.

Here is a summary of the steps to be taken for assigning parameters to the units and bus modules:

A	Define the device addresses (Ethernet addresses) on the SINEC-H1 bus
B	Generate the PLC program for group diagnosis. To do this, the job numbers for the link blocks must be defined in the SINEC-H1 CP.
C	Define the DISPLAY LOCATIONS for process error messages (in the setpoint data elements) and system and process control messages (in the CP 552-SYSID).
D	Decide on the plant name
E	Assign parameters to the LINK BLOCKS in the SINEC-H1 CP
F	Generate the configuration files of the display units with one CONFIGURATION per diagnostic processor, for which the PG acts as a display unit. The plant name of the respective configuration must be identical to the name in the CP 552-SYSID.
G	Create a path file with paths to the diagnostic processor and CPU. The name of the CPU path must be identical to the plant name.

2.3.2 Blocks on the CPU

For communication between the CPU and CP 552 you need the same blocks as those described in Section 2.2.1 for local error display. To exchange data via the SINEC-H1 bus you also need:

FB EMPFANG	Communication CPU / CP 535
FB SENDEN	Communication CPU / CP 535
FB HTB-SDIR	Communication CPU / CP 535
DB PUFFER	Communication CPU / CP 535 (parameters assigned in FB ANLAUF)

The function blocks **FB EMPFANG** and **FB SENDEN** must be linked into OB 1. In addition to this, you must also change the parameter assignment of **FB ANLAUF** in the organization blocks OB 20/21/22 (see also descriptions of standard function blocks for process error diagnosis).

2.3.3 Assigning Parameters to the CP 552

Using the programming package **COM 552**, you must specify the following in the SYSID of each CP 552:

- Plant identifier (max. 19 ASCII characters)
- Display locations for system messages from the CP 552
- Display locations for process control messages (only necessary when process control messages are being used).

Note:

If you enter **new values** in the SYSID, these are only accepted when you switch the CP 552 from "STOP" to "RUN."

2.3.4 Assigning Parameters to Display Units

Up to 16 diagnostic processors can send their messages to the same display unit. To create the required physical connections, you can use the **configuration editor** in the **COM 552** programming package.

The **CP 536** in the PG is assigned the following parameters to allow it to communicate with the CP 535 in the PLC via the bus in full duplex.

Plant:	corresponds to the plant identifier in the SYSID of the CP 552, to which the connection is to be established
Local TSAP-ID(S):	symbolic address for the SEND job on the CP 536
Local TSAP-ID(R):	symbolic address for the RECEIVE job on the CP 536
Unit:	specifies the binary identifier designating the display unit
Ethernet address:	bus address of the CP 535, assigned to the CP 552
Remote TSAP-ID(S):	symbolic address for the SEND job on the CP 535
Remote TSAP-ID(R):	symbolic address for the RECEIVE job on the CP 535
Symbols file:	if available, the symbols file belonging to the plant can be specified

The information you have generated with the configuration editor in COM 552 is stored in the **configuration file** on the PG. Information about the CP 535 can be found in the CP 535 manual (6ES5 998-0DG21).

Fig. 2-5 shows a **configuration file** with two complete paths to two diagnostic processors. Two links have been planned in the plant identifier in the SYSID of each diagnostic processor (SEND and RECEIVE channels).

Example: configuration file

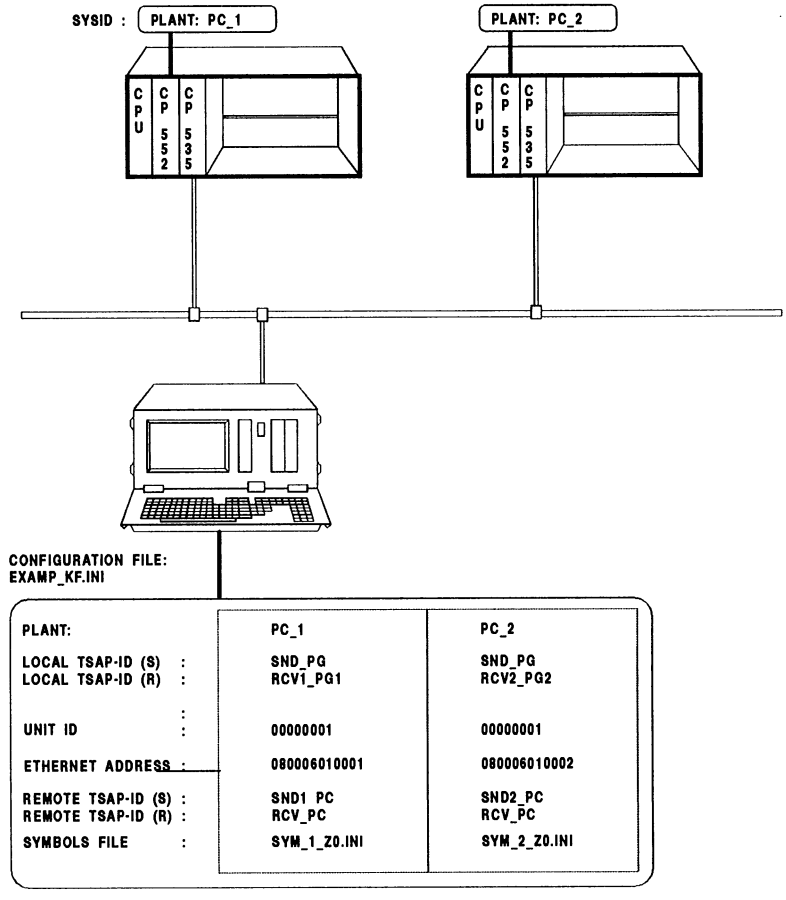


Fig. 2-5 Example of a configuration file - connection PG - CP 535 - CPU - CP 552

2.3.5 Assigning Parameters to the CP 535

For the example in Fig. 2-5, assign parameters to the CP 535 using the programming package **COM 535** as follows:

Plant 1:

Ethernet address (CP 535) = 080006010000 H (example)
Interface number (CP 535) = 0 (example)
Priority = 4 (must be specified)
Active/passive = passive

Job type	Job number ANR	TSAP-ID (local)	Ethernet Address (PG)	TSAP-ID (remote)
SEND	11	SND1_PG	0800060110EF	RCV1_PC1
RECEIVE	24	RCV_PG	000000000000	SND_PC

Plant 2:

Ethernet address (CP 535) = 0800060100001
Interface number (CP 535) = 0 (example)
Priority = 4 (must be specified)
Active/passive = passive

Job type	Job number ANR	TSAP-ID (local)	Ethernet Address (PG)	TSAP-ID (remote)
SEND	12	SND2_PG	0800060110EF	RCV1_PC2
RECEIVE	24	RCV_PG	000000000000	SND_PC

Further information about the programming package COM 535 can be found in the CP 535 manual. Information about assigning parameters to the standard function blocks for the CPU can be found in the description of the CP 552 standard function blocks.

The connection from the CP 552 to the display units for group diagnosis is via the CP 535. For each display unit (PG) you must program a separate SEND job with any job number on the CP 535. If the display unit requests error messages, the request is processed on the CP 535 by a single RECEIVE job with the non-specified remote Ethernet address 000000000000.

For the complete error request in group diagnosis, the RECEIVE job must always be the same.

2.3.6 Path File

Path to the CPU

To display the segment status, BSTACK and ISTACK you require a path via the CP 535 and the multiplexer (or KOR C with the S5-135U) to the CPU. To ensure that this path is automatically established by the display unit, create a path file with the **utility BUS SELECTION**. A path must be set up in the display unit for every CPU. The path name must be entered as the plant identifier in the SYSID of the corresponding CP 552.

If you wish to display:

- segment titles,
- statements comments and
- segment comments

you must enter the name of the program file belonging to each CPU when editing the path files using the function key **SUPP FUNCTION**.

Path to the CP 552

You can also establish a direct path to the CP 552, which can be used for the following functions:

- transferring setpoint data
- CP 552 functions RUN/STOP and DATE/TIME
- CP 552 information: INFORMATION and SYSID

In Fig. 2-6, PC_1 and PC_2 are paths to the CPUs. CP_1 and CP_2 are paths to the diagnostic processors.

Example: path file

Paths to CP 552 and to CPU for displaying network status, BSTACK und ISTACK via the PG channel of the CP 535

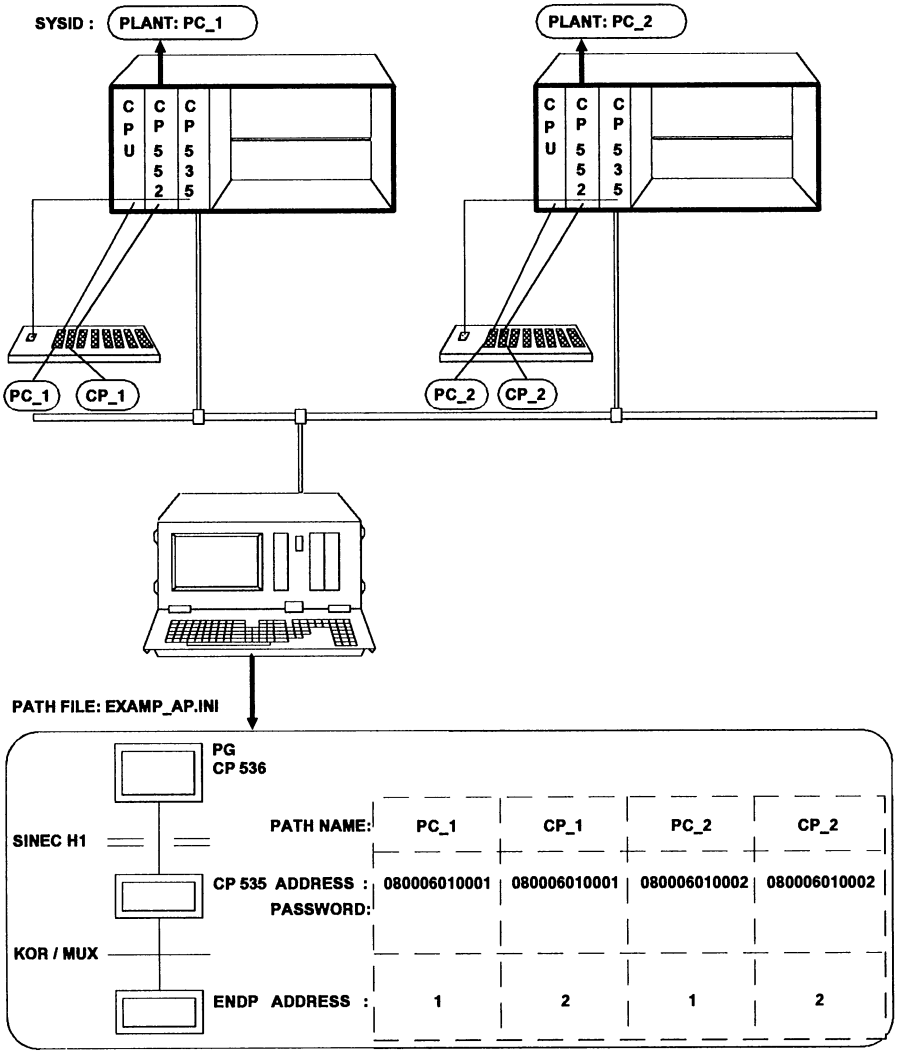


Fig. 2-6 Example of a path file

2.3.7 Assigning Parameters to the CP 536

Before starting the CP 536 in the PG for the first time, you must assign the appropriate Ethernet address. The **utility BUS SELECTION** is available for this purpose. Activate an existing path and press the keys "SINGLE" and "SYSID." You can now enter the address in the corresponding field.

2.3.8 Transferring Setpoint Data Elements and Displaying Process Errors

The generated setpoint data elements are transferred in COM 552 to the CP 552. The CP 552 must be in the STOP mode. Following this, switch the CP 552 to RUN. If you are unable to switch the CP 552 to RUN, you can display possible causes of error using the function "INFO" in the "CP 552 ONLINE".

Now exit the CP 552 ONLINE function. Select "GROUP" as the CONFIGURATION preset and enter the CONFIG FILE. Change to the CURRENT MESSAGES function. If the CP 552 is in the RUN mode, you can now display the process error messages. Further information about error display can be found in the User's Guide "COM 552 Programming Package".

2.3.9 Setting the Time/Date Centrally

If there are several programmable controllers with a CP 552 in your system, and if these are interconnected via SINEC H1, you can synchronize the hardware clocks of all the CP 552 modules. Using the programming package **COM 552** you only need to set the clock on **one** CP 552 in the master PLC. The synchronization is then carried out by the CPU of the master PLC and the slave PLCs. The CP 552 provides the following jobs for the CPU.

- **Set time/date: SEND DIRECT** with **job number 218**

- **Read time/date: RECEIVE DIRECT** with **with job number 218**

On the **CPU in the master PLC** you require the following

- one RECEIVE DIRECT 218
  to read the time/date from the CP 552
  (you can set the time/date on the CP 552 in COM 552)

- one SEND DIRECT 218
  for each CP 552 in the master PLC to which the time/date is transferred

- a SEND job to the CP 535 to transfer the time/date to the CP 535.

On the corresponding **CP 535** you require the following:

- a SEND job.

On the **CPU in the slave PLC** you require the following:

- one RECEIVE job
  to receive the time/date from the CP 535

- one SEND DIRECT 218 per CP 552
  to transfer the time/date to the CP 552s in the slave PLC

On the corresponding **CP 535** you require the following:

- one RECEIVE job
  to receive the time/date.

2.4 Displaying Process Control Messages

Fig. 2-7 shows the programming packages with which you plan the display of process control messages. For further information on planning process control messages, refer to the User's Guide "Displaying Process Control Messages" in this manual.

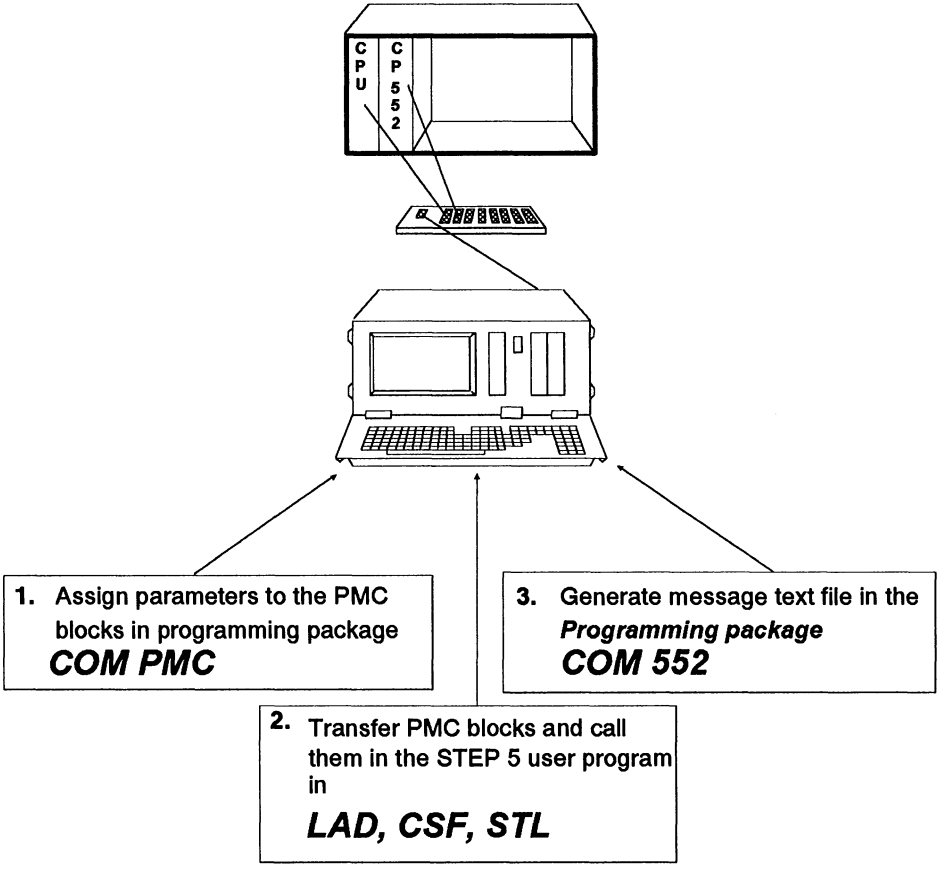


Fig. 2-7 Planning the display of process control messages

2.5 Start-up Procedure of the CP 552 in the PLC Rack

When all the parameters have been assigned as described in the previous sections, the only thing that remains is to synchronize the CP 552 with the CPU. The CP 552 is synchronized with the CPU as follows:

automatically	each time the CPU is run up when the mode selector on the CP 552 is set to "RUN"
manually	if the CPU is already in the "RUN" mode and you switch the CP 552 to "RUN" using the mode selector or by carrying out a cold restart with the PG (see instructions "Programming Package COM 552").

The handling of the jobs required to coordinate the CPU and CP 552 is performed by **standard function blocks for process error diagnosis with the CP 552 diagnostic processor**.

Process error diagnosis with the CP 552 begins immediately once the CP 552 is synchronized and data exchange has started. This takes 4 + n CPU cycles (in the best case, n = 0 with few setpoint data elements). In these n + 4 cycles, the CPU informs the CP 552 of the I/O configuration. The CP 552 informs the CPU which inputs, outputs and flags are to be transferred cyclically. **Only process elements started after these n + 4 cycles can be monitored.**

 **Note:**

The **start condition** is checked when you start the CP 552 (mode selector from "STOP" to "RUN").

2.6 Calling and Assigning Parameters to the Blocks on the CPU

This section describes the most important operations you must carry out in your STEP® 5 user program. Detailed programming instructions for the blocks required for process error diagnosis can be found in your programmable controller manual. The FB ANLAUF call mentioned in Section 2.2.1 in the organization blocks OB 20/21/22 must be structured as follows:

OB 20 (or 21 or 22)

```
      : JU  FB 44
NAME  : DPANLAUF
SSDP  :      KF+4      Interface no. CP 552
ANZZ  :      KS  N     Group diagnosis via SINEC-H1
SSCZ  :      KF +0     Interface no.
                       SINEC CP (CP 535)
BLGR  :      KF +4     Frame size for transfer
                       via SINEC H1
DBEA  :      KF +211   DB-EA
DBPU  :      KF +209   DB Buffer
MELD  :      KS  N     Display of Process control Messages
ANZL  :      KS  N     Display also via CP 527
SSCL  :      KF +0     Interface no. CP 527
```

OB 1

```
      : JU  FB 43
NAME  : KOMCP552
BLCK  :      KF +210   DB Block
SAMM  :      KF +208   DB Sammel (only for the S5-135U)
```

If you would like the error identifiers with the CP 552 to be evaluated by the CPU, you must also include a handling block call of the following type in your STEP® 5 user program:

```
      PB 20                SEGMENT 2

      : JU  FB 121         ;  FB181 with S5-150U, FB 121 with S5-135U and
                           ;  155U, FB 245 with S5-115U
NAME  : RECEIVE
SSNR  :      KY 0,4        ;  interface number set on the CP 552
A-NR  :      KY 0,200      ;  job number 200
ANZW  :      FW 70
ZTYP  :      KS DB
DBNR  :      KY 0,70       ;  dest. address for the error identifiers
ZANF  :      KF+0          ;  at start of DB 70
ZLAE  :      KF+3          ;  length 3 words
PAFE  :      FY 69
      : BE
```

Three error identifiers are stored in words at the start of DB 70 (as described in Section 2.2.2). Remember that these error identifiers are overwritten when the next RECEIVE DIRECT 200 is called.

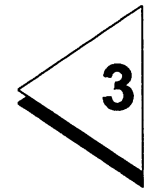
A general description of how to call the handling blocks can be found in the descriptions for the handling blocks.

The **FB RECEIVE** call takes the following form:

```
      : JU  FB 45

NAME  : RECEIVE
ANRZ  :      KF 24      RECEIVE function on CP 535

ANRL  :      KF 0       RECEIVE function on CP 527
```

The **FB SENDEN** call takes the following form:

```
      : JU  FB 46
NAME  : SENDEN
ANZ1  :      KY 11,0    Job numbers for the send job to display unit;
                        right byte: irrelevant.
                        Setpoint data elements with the display
                        location: xxxx xxx1  (x signifies don't care (0 or 1))
                        are sent with this A-NR

ANZ2  :      KY 12,0    Job numbers for the send job to display unit;
                        right byte: irrelevant.
                        Setpoint data elements with the display
                        location: xxxx xx1x are sent with this A-NR

ANZ3  :      KY 0,0
ANZ4  :      KY 0,0     Display units
ANZ5  :      KY 0,0     3 to 8
ANZ6  :      KY 0,0     not connected
ANZ7  :      KY 0,0
ANZ8  :      KY 0,0
```



Appendix - Examples of Error Display on the PG

Contents

This section provides examples of error displays on the PG. Fig. 3-1 shows the CURRENT MESSAGES screen form with the **error overview**.

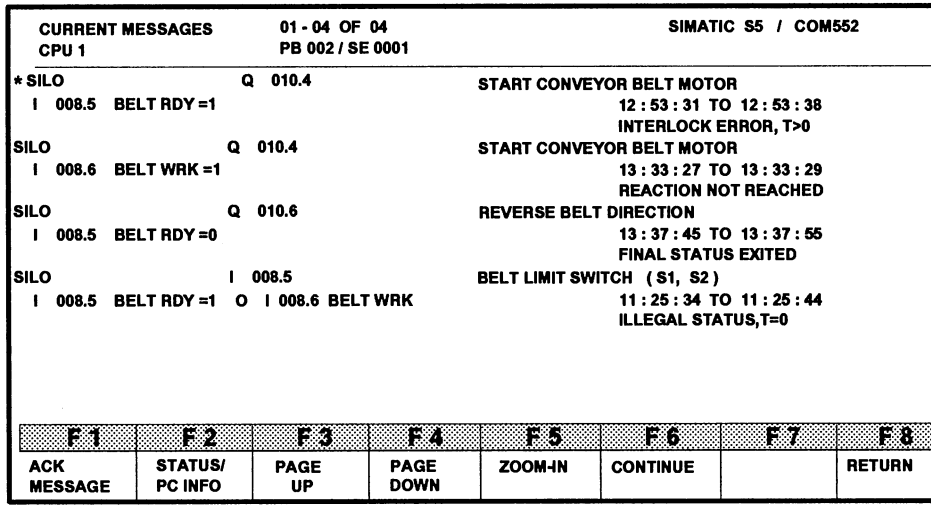


Fig. 3-1 Example of the error overview

Error 1: Interlock diagnosis - monitoring the process requirement
Error 2: Action diagnosis - monitoring the process function
Error 3: Reaction diagnosis - monitoring the final process status
Error 4: Monitoring the static signals

If you press **F5 zoom-in**, you obtain the zoom-in of the error message you have selected (Figs. 3-2, 3-3, 3-4 and 3-5). For further information refer to the User's Guide "COM 552 Programming Package".

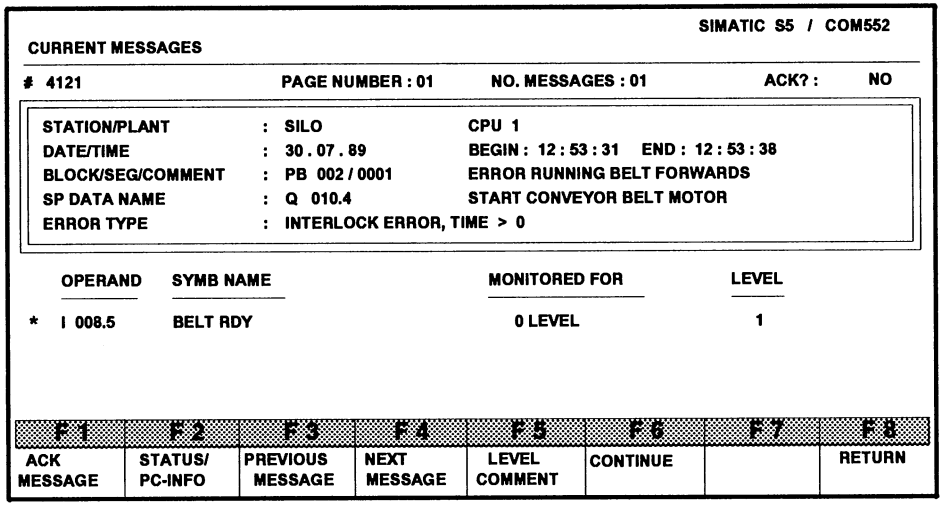


Fig. 3-2 Example of zoom-in representation
Action diagnosis: monitoring the process function

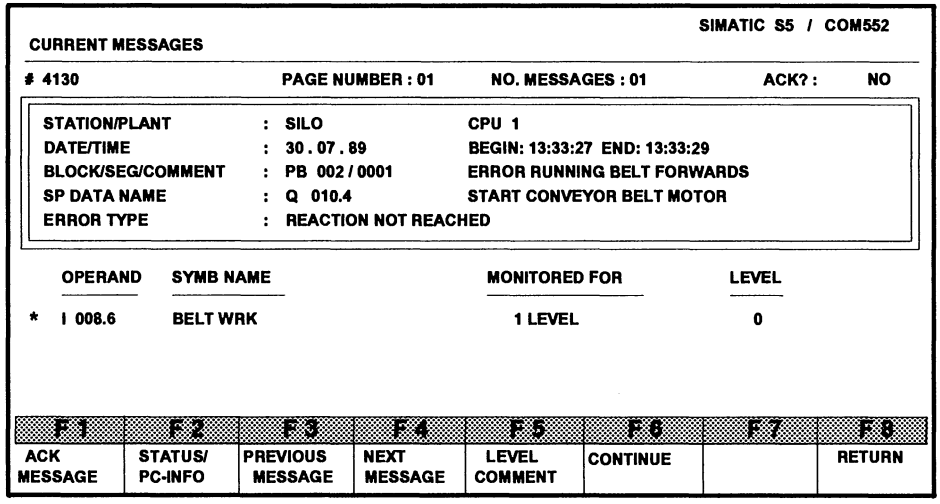


Fig. 3-3 Example of zoom-in representation
Reaction diagnosis: Monitoring of the process function

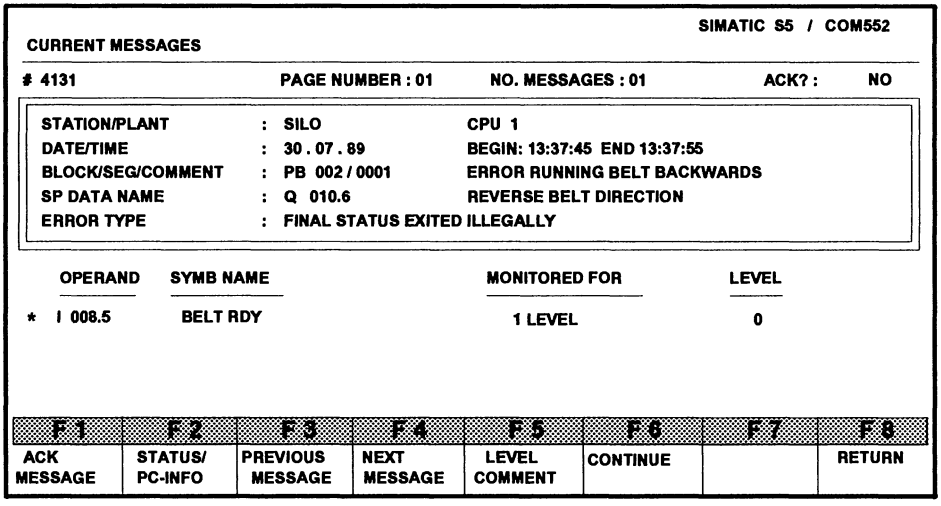


Fig. 3-4 Example of zoom-in representation
Monitoring the final process status

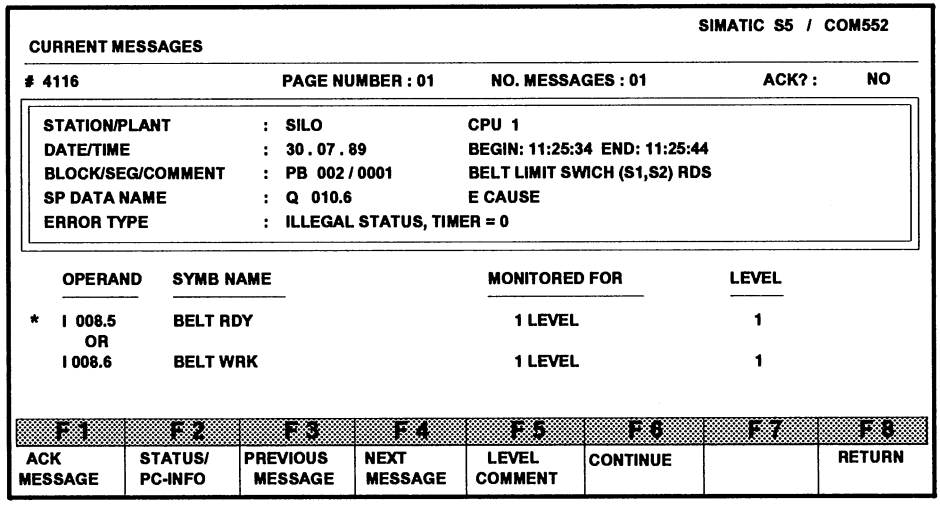


Fig. 3-5 Example of zoom-in representation
Monitoring static signals

Index

R

S

T

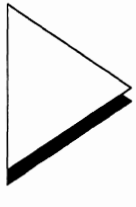
SIEMENS

SIMATIC S5

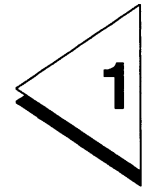
Process Error Diagnosis with the CP 552

Practical Example

C79000-B8576-C671-05



Contents



Introduction

Contents

This example of an application is based on the model of a **carwash** and describes the following:

- how to create the STEP® 5 user program with the corresponding **setpoint data** for process error diagnosis
- how to start the process error diagnosis.

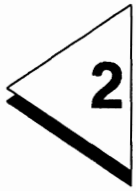
For this example, you require the following **hardware:**

- one S5-135U programmable controller
- one CPU 928 or CPU 922 (R processor)
- one PG 685 programmer
- one CP 552 diagnostic processor
- one multiplexer (PG-MUX)
- one digital input module, e.g. 6ES5 420 - 4UA11
- one digital output module, e.g. 6ES5 441 - 4UA11
- one simulator
- one cable connector (connection PG/MUX) type 6ES5 731-1...
- one cable connector (connection MUX/PLC) type 6ES5 731-1...
- one cable connector (connection MUX/CP 552) type 6ES5 731-1...

You also require the following **software:**

- S5-DOS from version 3.x onwards
- the LAD, CSF, STL package
- the DIAGNOSIS software package
- standard function blocks for process error diagnosis with the CP 552 diagnostic processor
- handling blocks for the S5-135U

Further information about order numbers can be found in the ordering data.



Description of the Process

Contents

In this example of an application, you will generate the setpoint data of a carwash for process error diagnosis with the CP 552.

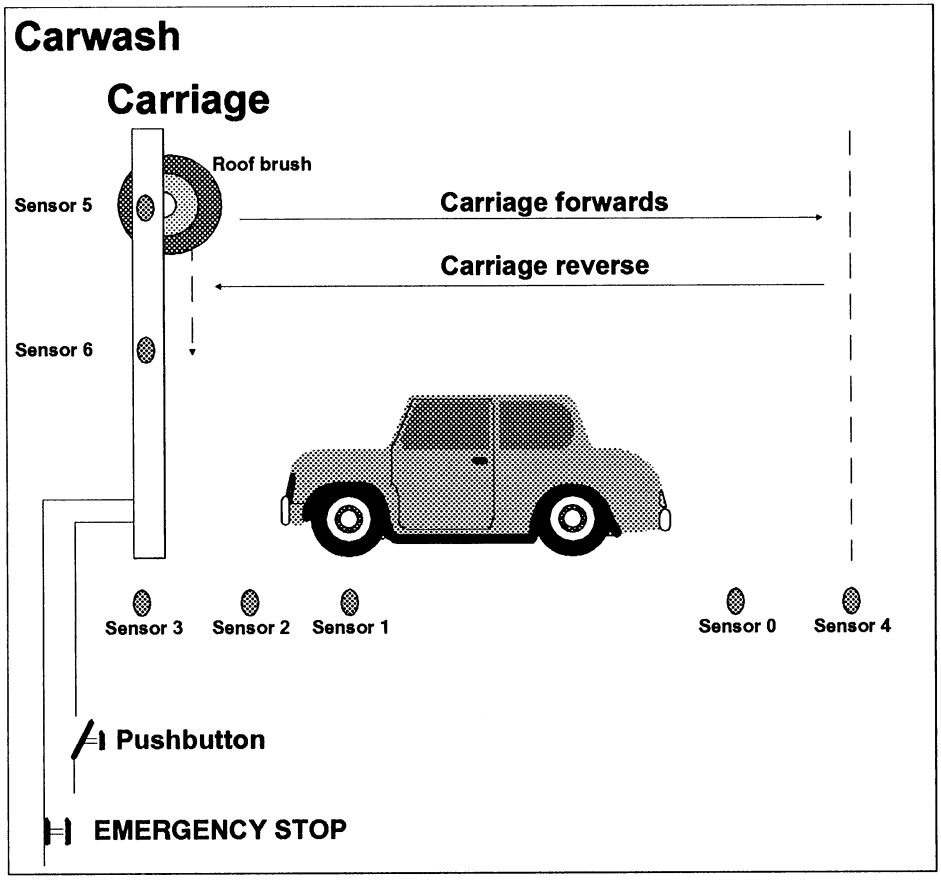


Fig. 2-1 Carwash

In the carwash, the **carriage** supporting the brushes moves between two end positions. Once the carriage reaches one of these end positions, this is signalled by **sensor 3** (S3) or by **sensor 4** (S4). The carriage movement is started by the **pushbutton**.

The correct position of the vehicle is checked by **sensor 1** (S1) and **sensor 2** (S2). The front of the vehicle must be between these two sensors. If the vehicle is too long, this is detected by **sensor 0** (S0). In this case, the carriage will not move.

The carriage motor can only start when a certain combination of interlocks is fulfilled:

- the EMERGENCY STOP switch is not active,
- the vehicle is in the correct position,
- the vehicle is not too long and
- the carriage reverse motor is not running.

If the interlock conditions are fulfilled and the pushbutton is pressed, the carriage starts moving **(carriage forward motor)** and reaches sensor 4 (S4) after **60 seconds**. When the carriage has reached sensor 4, the reverse movement is started **(carriage reverse motor)**. After a further **60 seconds**, sensor 3 must detect the end of the reverse movement.

The carriage supports three brushes, one on each side of the carriage and one on the cross piece **(roof brush)**. This roof brush can be moved vertically between **sensor 5** (S5) and **sensor 6** (S6). It should be in the upper position (sensor 5) when the carwash is first started or restarted. If it is not in this position, the roof brush up motor should be activated.

The assignment of the addresses in this example is as follows:

Inputs:

Pushbutton	I 3.0	
Sensor 0	I 2.0	Vehicle too long
Sensor 1	I 2.1	Vehicle forwards
Sensor 2	I 2.2	Vehicle reverse
Sensor 3	I 2.3	End position carriage back
Sensor 4	I 2.4	End position carriage front
Sensor 5	I 2.5	End position brush up
Sensor 6	I 2.6	End position brush down
EMERGENCY STOP	I 2.7	

Outputs:

Carriage forward motor	Q 2.0
Carriage reverse motor	Q 2.1
Roof brush up motor	Q 2.2

Flags:

Start-up flag	F 5.0

Process error diagnosis with the CP 552 is intended to detect and display process errors. Process errors could, for instance, be caused by signal line breaks, defect sensors or actuators and missing interlock signals. The CP 552 recognizes the process errors by comparing the setpoint data and the actual data from the process. The actual data is the current process image, which the CPU of the programmable controller provides for the CP 552. The setpoint data define the error-free functioning of the process and must be generated by you and transferred to the CP 552.

Before you can generate the setpoint data however, you must divide your process into **process elements**. You then generate at least one setpoint data element for each process element. Each process element is a complete activity. The more process elements the process is divided into for process error diagnosis, the more setpoint data elements will have to be generated, resulting in a more detailed diagnosis.

Within your process, you can monitor dynamic functions and static statuses (see Fig. 2-2). For the dynamic functions, you must identify the process requirement, the process function and the final process status for each process element.

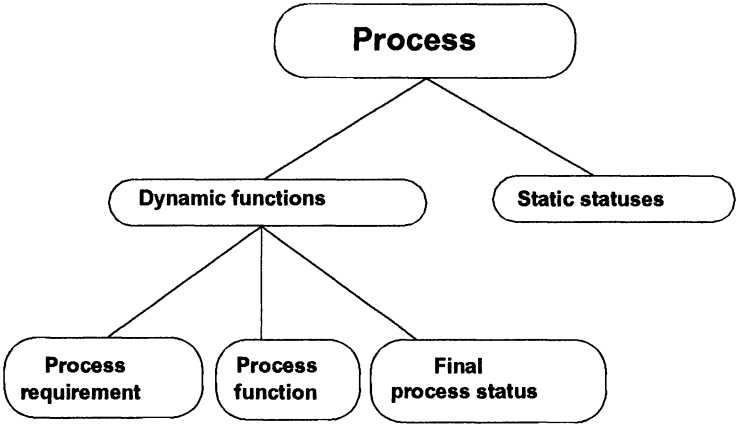


Fig. 2-2 Process

In this example of a carwash, the two process elements are as follows:

- **Carriage forwards**
- **Carriage backwards**

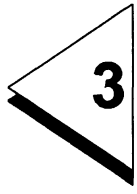
However, to obtain a detailed diagnosis, the carriage movements both forwards and backwards should be divided into further process elements.

The following pages describe the next steps as follows:

- how to break down the carriage forwards and backwards movements into further process elements
- how to generate the setpoint data elements to monitor the dynamic functions
- how to generate the setpoint data elements to monitor the statuses

 **Note:**

When monitoring dynamic processes, you define the **error-free** process function in the setpoint data elements. To monitor statuses, you define the **illegal** status.



Setpoint Data Elements

Contents

To obtain a detailed diagnosis, break down the **carriage forwards** and **carriage backwards** movements into further process elements. You then generate the setpoint data elements for these process elements.

3.1 Setpoint Data Elements for Carriage Forwards

This process element is divided into the following process elements:

- **carriage starts to move in a forwards direction**
- **carriage moves from sensor 3 to sensor 4.**

For each of these process elements, you generate one setpoint data element. You also generate a setpoint data element to monitor the statuses.

The setpoint data element for "carriage starts to move in a forwards direction" monitors whether

- the vehicle is in the correct position and remains there
- the carriage begins to move (leaves the end position carriage back, sensor 3) after the pushbutton is pressed.

The setpoint data element "carriage moves from sensor 3 to sensor 4" monitors whether

- sensor 4 is reached within 60 seconds.

The setpoint data element for monitoring the statuses monitors whether

- certain sensors and signal lines are functioning correctly.

3.1.1 Carriage starts moving in a Forwards Direction

Since this is a dynamic function, you must recognize the following parts of the process element:

1. **Process requirement**
2. **Process function**
3. **Final process status**

1. Process requirement

Considerations:

- What is the actual trigger for the process element and which interlock conditions must be met?
- After the trigger becomes active, what time must be allowed for the action to take place?
- What is the action?

The process element "carriage starts moving in a forwards direction" is triggered by the positive (rising) edge of the pushbutton signal. When the pushbutton is pressed, the carriage forwards motor must be activated immediately, i.e. in the same cycle.

Trigger:	Pushbutton = P	(positive edge)
Trigger time:	0 seconds	
Action:	Carriage forwards motor = 1	

Before the motor can start, however, the following interlock conditions must be met:

- the EMERGENCY STOP is not active
- the carriage is at the end position (sensor 3)
- the vehicle is in the correct position
- the vehicle is not too long
- the carriage reverse motor is not switched on

The interlocks to be monitored must be specified for the action in this example. The more interlocks specified, the more exact the automatically generated error message will be:

Action (with interlocks):

	Carriage forwards motor = 1	
AND	EMERGENCY STOP = 1	Not active
AND	Sensor 3 = 1	Carriage at end position
AND	Sensor 1 = 1	Vehicle in correct
AND	Sensor 2 = 0	position
AND	Sensor 0 = 0	Vehicle not too long
AND	Carriage reverse motor = 0	

 **Note:**

If you specify the interlocks with the action, the monitoring is terminated as soon as an interlock condition is no longer fulfilled. If you do not require this, the interlocks should be assigned their own setpoint data element.

2. Process function

Considerations:

- Which reaction must occur after the action?
- After the action has started, how much time must be allowed for the reaction to occur?
- What type of movement is involved? Must the action remain valid until the reaction occurs (motive) or is the triggering of the activity sufficient (pulse-dependent) or is a combination of both required?

Once the carriage has started moving, it must leave sensor 3 within 0.5 seconds (reaction). The end position can only be reached when the carriage motor remains switched on until the reaction occurs.

Movement, motive	(x)
Monitoring time:	500 ms
Reaction:	Sensor 3 = 0

3. Final process status

Considerations:

- Must the reaction remain valid once it has occurred?
- If yes, which condition stops the monitoring and which tolerance time is allowed?

The monitoring should only stop when the reverse movement starts. The reaction can, however, disappear for 100 ms, without an error message being generated (for example, if the limit switch bounces, it will not lead to an error message).

Reaction monitoring	(x)
Tolerance time:	100 ms
Stop condition:	Carriage reverse motor = 1

The setpoint data element for the process element "**carriage starts moving in a forwards direction,**"

Movement, motive	(x)	
Reaction monitoring	(x)	
Trigger:	I 3.0 = P	; Pushbutton; positive edge triggers the action;
Trigger time:	0 * (100ms)	
Action:	Q 2.0 = 1	; Carriage forwards motor on
A	I 2.7 = 1	; EMERGENCY STOP not active
A	I 2.1 = 1	; Vehicle is in
A	I 2.2 = 0	; correct position
A	I 2.0 = 0	; Vehicle not too long
A	I 2.3 = 1	; End position carriage back
A	Q 2.1 = 0	; Carriage reverse motor off
Monitoring time:	5 * (100ms)	
Reaction:	I 2.3 = 0	; End position carriage back has been left
Tolerance time:	1 * (100ms)	
Stop condition:	Q 2.1 = 1	

3.1.2 Carriage moves from Sensor 3 to Sensor 4

The carriage must reach end position sensor 4 within 60 seconds.

1. Process requirement

In this case, no interlock diagnosis is carried out, since the interlocks have already been taken into account in the setpoint data element for the process element "carriage starts moving in a forwards direction". The trigger and trigger time are also ignored.

2. Process function

When the carriage forwards motor is started, sensor 4 must be reached within 60 seconds.

Action:	Carriage forwards motor = 1
Monitoring time:	60 seconds
Reaction:	Sensor 4 = 1

This function is also motive, since the action (carriage forwards motor) must remain valid until the reaction occurs (sensor 4 = 1).

Movement, motive	(x)

3. Final process status

The reaction should be monitored until the stop condition (carriage reverse motor) occurs. A tolerance time of 100 ms is advisable to prevent switch bounce leading to an error message.

Reaction monitoring	(x)
Tolerance time	100 ms
Stop condition	Carriage reverse motor = 1

The setpoint data element for the process element "**carriage moves from sensor 3 to sensor 4**":

Movement, motive	(x)	
Reaction monitoring	(x)	
Trigger:	-----	;Trigger and trigger time are not specified here, since interlock diagnosis was already carried out in the last setpoint data element
Trigger time:	--- * (100ms)	
Action:	Q 2.0 = 1	; Carriage forwards motor on
Monitoring time:	600 * (100ms)	
Reaction:	I 2.4= 1	; Final position carriage front
Tolerance time:	1 * (100ms)	
Stop condition:	Q 2.1 = 1	; Carriage reverse motor

3.1.3 Monitoring Statuses

Illegal statuses include the following:

- the vehicle moves from the correct position while the carriage is moving or sensor 1 and sensor 2 are defective (both sensors are active or sensor 1 is not active).

  Illegal status:

	Carriage forwards motor = 1
AND	Sensor 1 = 1
AND	Sensor 2 = 1
OR	Carriage forwards motor = 1
AND	Sensor 1 = 0

- while the carriage is moving, the signal "vehicle too long" is received (sensor 0 defect or vehicle is rolling backwards)

  Illegal status:

	Carriage forwards motor = 1
AND	Sensor 0 = 1

- Sensors 3 and 4 are active simultaneously

  Illegal status:

	Sensor 3 = 1
AND	Sensor 4 = 1

100 ms is sufficient tolerance time to avoid error signals resulting from switch bouncing.

The setpoint data element for **monitoring statuses**:

Illegal statuses:

```
      Q 2.0 = 1      ; carriage forwards motor
A     I 2.1 = 1      ; vehicle no longer in
A     I 2.2 = 1      ; correct position
O     Q 2.0 = 1      ;
A     I 2.1 = 0      ;
O     Q 2.0 = 1      ;
A     I 2.0 = 1      ; vehicle too long
O     I 2.3 = 1      ; end position carriage back
A     I 2.4 = 1      ; end position carriage front
```

Tolerance time: 100 ms

3.2 Setpoint Data Elements for Carriage Backwards

The setpoint data elements for carriage backwards are generated in the same way as the setpoint data elements for carriage forwards. Break down the carriage backwards movement into process elements as follows:

- **Carriage starts moving in the reverse direction**
- **Carriage moves from sensor 4 to sensor 3**

You also generate a setpoint data element for monitoring the static statuses.

3.2.1 Carriage starts moving in the Reverse Direction

1. Process requirement

The reverse movement (carriage reverse motor) is triggered when sensor 4 is reached, i.e. as soon as the carriage has reached the front.

Trigger:	Sensor 4 = 1
Trigger time:	0 seconds
Action:	Carriage reverse motor = 1

The carriage reverse motor is, however, only started when the following interlock conditions are fulfilled:

- EMERGENCY STOP is not active
- the vehicle is in the correct position
- the vehicle is not too long (sensor 0 has not responded)
- the carriage forwards motor is not switched on

Action (with interlocks):

	EMERGENCY STOP = 1
AND	Sensor 0 = 0
AND	Sensor 1 = 1
AND	Sensor 2 = 0
AND	Sensor 3 = 0
AND	Carriage forwards motor = 0

2. Process function

When the carriage reverse motor is started, the carriage must leave sensor 4 within 0.5 seconds. The type of movement is motive, since the motor must remain switched on until the end position is reached.

Movement, motive	(x)
Monitoring time:	500 ms
Reaction:	sensor 4 = 0

3. Final process status

The reaction sensor 4 = 0 must be active until the end position carriage back is reached. The end position carriage back should be the stop condition for the whole monitoring (reaction monitoring). Specify a tolerance time of 100 ms (if the limit switch bounces, this should not lead to an error message).

Reaction monitoring	(x)
Tolerance time:	100 ms
Stop condition:	end position carriage back = 1

The setpoint data element for the process element "carriage starts moving in the reverse direction":

Movement, motive	(x)	
Reaction monitoring	(x)	
Trigger:	I 2.4 = 1	; End position carriage front
Trigger time:	0 * (100ms)	
Action:	Q 2.1 = 1	; Carriage reverse motor on
A	I 2.7 = 1	; EMERGENCY STOP not active
A	I 2.1 = 1	; Vehicle is in
A	I 2.2 = 0	; correct position
A	I 2.0 = 0	; Vehicle not too long
A	Q 2.0 = 0	; Carriage forwards motor off
Monitoring time:	5 * (100ms)	
Reaction:	I 2.4 = 0	; End position carriage front has been left
Tolerance time:	1 * (100ms)	
Stop condition:	I 2.3 = 1	

3.2.2 Carriage moves from Sensor 4 to Sensor 3

The setpoint data element should monitor whether the carriage arrives at sensor 3.

1. Process requirement

Trigger and trigger time do not need to be monitored here, since these are taken into account in the setpoint data element for "carriage starts moving in the reverse direction."

2. Process function

When the carriage reverse motor starts, sensor 3 should be reached within 60 seconds. The movement is motive.

Movement, motive:	(x)
Action:	Carriage backwards motor = 1
Monitoring time:	60 seconds
Reaction:	Sensor 3 = 1

3. Final process status

The reaction should remain active until the carriage forwards motor is activated (reaction monitoring). Specify 100 ms as the tolerance time (bounce time).

Reaction monitoring	(x)
Tolerance time:	100 ms
Stop condition:	Carriage forwards motor = 1

The setpoint data element for the process element **"carriage moves from sensor 4 to sensor 3"**:

Movement, motive	(x)	
Reaction monitoring	(x)	
Trigger:	------	
Trigger time:	--- * (100ms)	
Action:	Q 2.1 = 1	; Carriage reverse motor
Monitoring time:	600 * (100ms)	
Reaction:	I 2.3 = 1	; End position carriage back reached
Stop condition:	Q 2.0 = 1	

3.2.3 Monitoring Statuses

It is advisable to monitor incorrect statuses in the reverse direction as well. The vehicle must not move away from the correct position as long as the carriage motor is switched on.

Incorrect status:

	Carriage reverse motor = 1	
AND	Sensor 1 = 1	Vehicle driven too far; both sensors respond
AND	Sensor 2 = 1	
OR	Carriage reverse motor = 1	
AND	Sensor 1 = 0	Vehicle too far back; sensor 1 does not respond
OR	Carriage reverse motor = 1	
AND	Sensor 0 =1	Vehicle too long

The tolerance time should be 100 ms.

The setpoint data element for **monitoring the statuses:**

Illegal status:

```
      Q 2.1 = 1      ; Carriage reverse motor
A     I 2.1 = 1      ; Vehicle not in
A     I 2.2 = 1      ; correct position
O     Q 2.1 = 1
A     I 2.1 = 0
O     Q 2.1 = 1
A     I 2.0 = 1      ; Vehicle too long
```

Tolerance time: 100 ms

3.3 Setpoint Data Elements for Start-up

Following a cold or warm restart of the CPU, the carriage and roof brush must be in the start position (carriage back, brush up). If the carriage and roof brush are not in the start position, they must be moved to this position. The start positions carriage back and roof brush up must be reached within 60 seconds.

Action:

	Carriage reverse motor = 1
AND	EMERGENCY STOP = 1
OR	Roof brush up motor = 1
AND	EMERGENCY STOP = 1

Monitoring time: 60 seconds

Reaction:

	Sensor 3 = 1
AND	Sensor 5 = 1

Since this monitoring is only to be carried out following a cold or warm restart, it is released by the start-up flag in the STEP® 5 user program. This is set in the start-up OBs (OB 20, OB 21 and OB 22). The monitoring should be stopped when the carriage forwards motor is started.

Release monitoring: Start-up flag = 1

Stop condition: Carriage forwards motor = 1

The setpoint data element for the function **"move system to start position following a cold or warm restart"**:

Movement, motive	(x)
Reaction monitoring	(x)
Start condition	(x)
Release monitoring:	F 5.0 = 1
Action:	Q 2.1 = 1
A	I 2.7 = 1
O	Q 2.2 = 1
A	I 2.7 = 1
Monitoring time:	60 seconds
Reaction:	I 2.3 = 1
A	I 2.5 = 1
Tolerance time:	100 ms
Stop condition:	Q 2.0 = 1

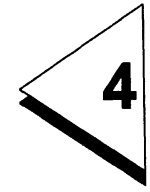
This monitoring is only to be carried out **after a cold or warm restart of the CPU**. The start-up flag in the STEP® 5 user program is therefore used as the release.

Note:

Following each **cold or warm restart** of the CP 552 the **start condition** is checked by the CP 552, i.e. it checks whether the conditions of the reaction field are fulfilled (assuming that the setpoint data element is released and start condition has been selected).

Note:

If the CP 552 is not inserted or is defective, the STEP® 5 user program cannot be started. You can, however, start the STEP® 5 user program if you include a time monitoring function of 1 second in OB 1 to monitor data bit 57.1 in DB PARAM.



Hardware

Contents

4.1 System Configuration with the S5-135U

The programmer can access the CPU and the CP 552 via the multiplexer (PG-MUX).

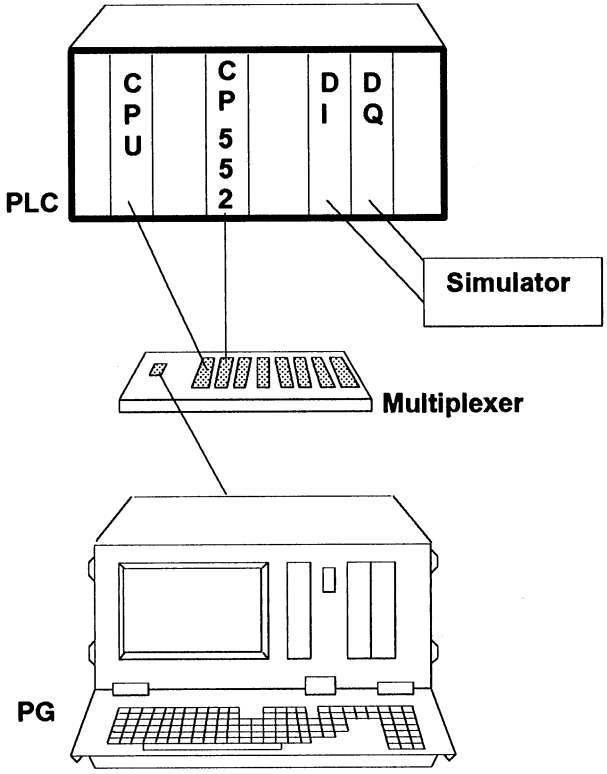


Fig. 4-1 System configuration

4.2 Programmer

For setpoint data generation, setpoint data transfer and error display, you must load the software package DIAGNOSIS in the programmer. Select user area 0 on your PG:

0:

Insert diskette 1 (1/2) and then diskette 2 (2/2) in the floppy disk drive (left-hand drive) and transfer the contents to the hard disk.

PIP B:=A:*.*[R V]

To be able to access the files from any user area, you must declare the files as system files. It is also advisable to protect the files from being written to or deleted by assigning the read-only attribute.
SET S5??D02X.*[SYS RO]
SET S5O?S0GX.*[SYS RO]

Exit user area 0, which should be reserved for system files, by entering the following:

n: (n = required user area)

Connect the cable connector (connection PG / MUX) to the multiplexer (see Fig. 4-1).

You must now create the path file for path selection via the multiplexer. To do this, call the STEP® 5 BASIC PACKAGE with the following command:

S5

When the S5 screen form SELECT PACKAGE appears on the screen, press **F2 UTILITY**. By pressing **F1 BUS SEL** the BUS SELECTION program is activated. You can now enter the PATH NAME and PATH FILE.

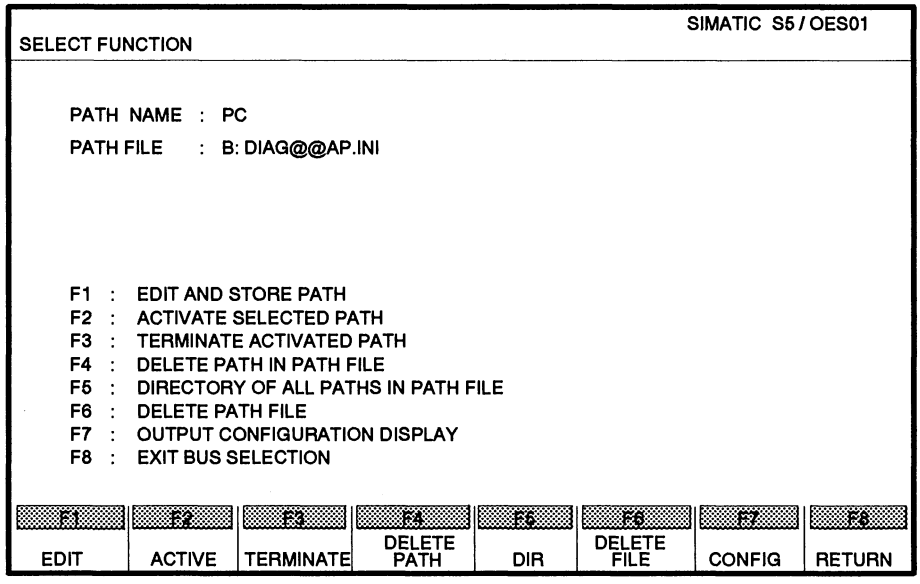


Fig. 4-2

As the PATH NAME specify the following:

PC

As PATH FILE enter the following:

B:DIAG@@AP.INI

Press **F1** to call the editor with which you can generate a path.

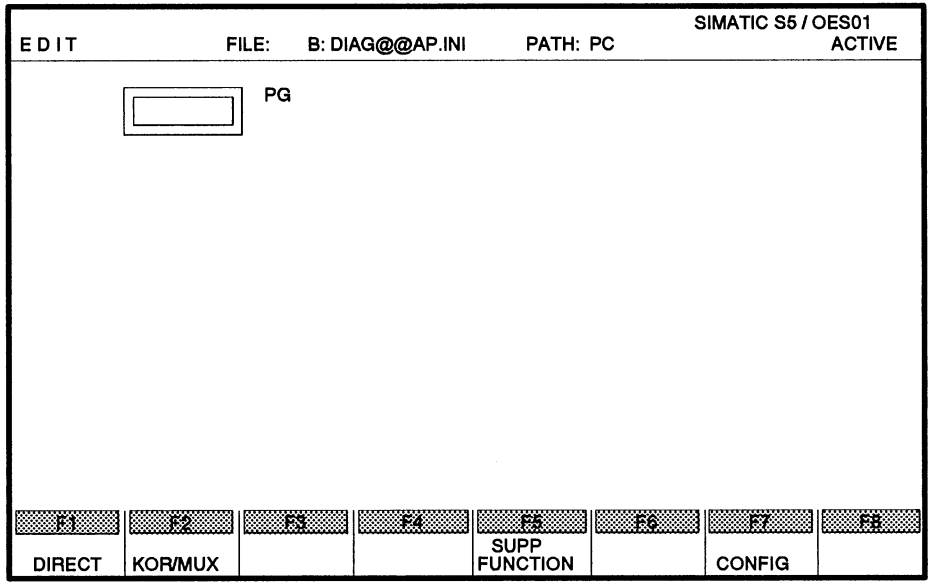


Fig. 4-3

Press **F2 KOR/MUX** in the EDIT screen form, and then press **F1 ENDP**.

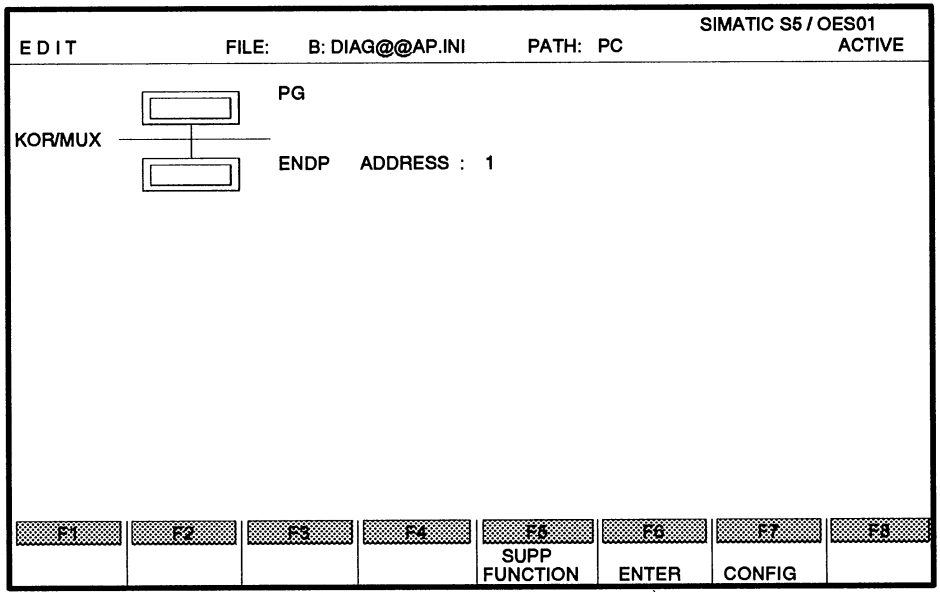


Fig. 4-4

This path is to have the following **ENDP ADDRESS**:

1

Press **F6 ENTER**, to enter the data. The basic screen form for BUS SELECTION is now displayed.

Following the same procedure, generate another path with the PATH NAME

DIAGPROCESSOR01

which also has the ENDP ADDRESS

1

This path must be created, to be able to call the ISTACK and the segment status automatically during error display in the programming package COM 552. Any path name can be selected, however, **it must match the plant designation in the SYSID of the CP 552**.

You must now create the path to the CP 552. Enter the following PATH NAME:

CP552

Then press **F1** again for the path editor. Carry out the same procedure once again, except that this time the ENDP ADDRESS should be

2

Enter the data with **F6** and exit the screen form with **F8**.

4.3 S5-135U

Configure the S5-135U programmable controller following the guidelines in the manual. Insert the correctly set digital modules (address 0) in the peripheral slots in the S5-135U.

4.3.1 CP 552

There are no settings required on the CP 552, since the module is correctly set when it is supplied. The preset for the even interface number when supplied is 0, i.e. all the switches on the DIP switch J56 are set to OFF. In this example, you will use this preset.

Insert the CP 552 in a suitable slot in your programmable controller (see Instructions "Diagnostic Processor CP 552") and connect it to the second MUX output interface with the second cable connector (connection MUX/CP 552).

4.3.2 CPU

Connect the multiplexer with the CPU, connect the cable connector (connection MUX/PLC) to the first MUX output interface (see Fig. 4-1).

Call the LAD, CSF, STL package in the S5 screen form SELECT PACKAGE. In the PRESETS screen form, you should enter a PROGRAM FILE, the PATH NAME specified above and the PATH FILE. Select "WITH DIAG."

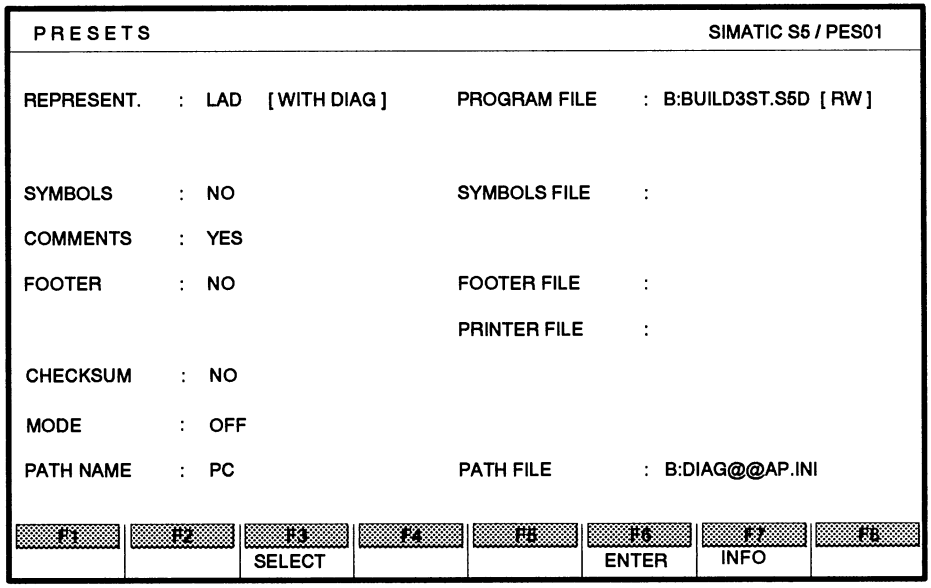
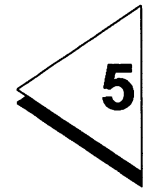


Fig. 4-5

You then obtain the information that the path is active. Now clear all the programs from the RAM area by carrying out an overall reset of the CPU.



STEP® 5

User Program and Setpoint Data Elements

Contents

The following sections describe how you generate the STEP® 5 user program and the corresponding setpoint data elements for the carwash using the LAD, CSF, STL package.

Within the LAD, CSF, STL package, you also transfer the standard function blocks and the data blocks as well as the handling blocks from the diskette to the CPU. First transfer all standard FBs/DBs and then the handling blocks.

A simple STEP® 5 user program is adequate for this example. First enter **FB 10**. This function block will bring the carriage and roof brush to their initial position following a cold or warm restart.

```
FB 10                    LEN=16

SEGMENT 1                0000

NAME :START

0005  :A    I 2.7       EMERGENCY STOP NOT ACTIVE
0006  :AN   I 2.5       MOVE ROOF BRUSH UPWARDS TO
0007  :=    Q 2.2       INITIAL POSITION
0008  :
0009  :A    I 2.7       EMERGENCY STOP NOT ACTIVE
000A  :AN   I 2.3       MOVE CARRIAGE BACK TO
000B  :=    Q 2.1       INITIAL POSITION
000C  :BE
```

Once you have entered the command "BE" the screen form PROCESS ELEMENT MONITORING for process error diagnosis appears on the screen.

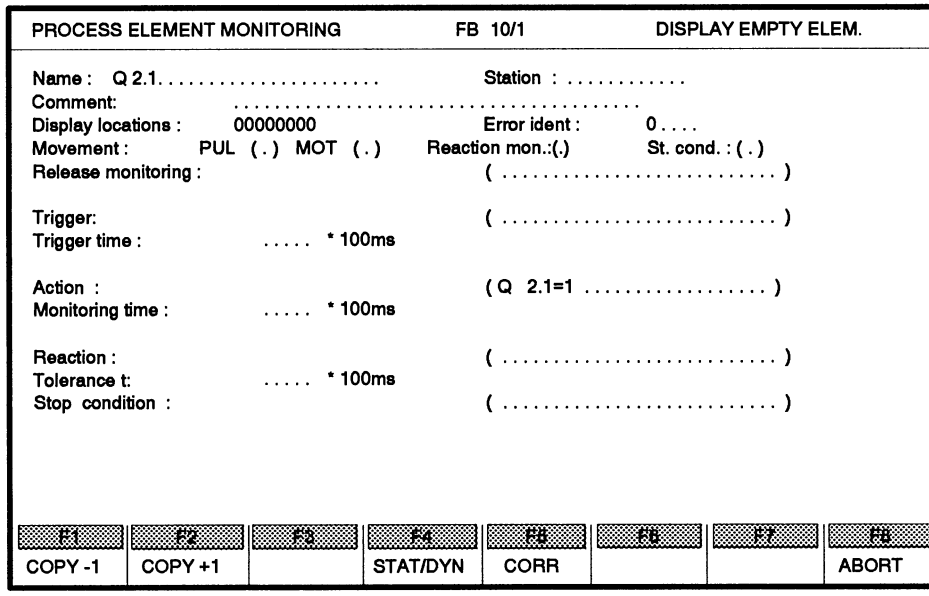


Fig. 5-1

You can fill in this screen form by pressing **F5 CORR**, the cursor is then positioned on the first input field. The field for the action is preset with the last term in the corresponding segment. The first term of the action field is automatically entered in the field for the name of the setpoint data element. Fill in the fields of the PROCESS ELEMENT MONITORING screen form as follows:

Enter the designation of the station and a comment, so that you can allocate the error messages to the process elements better. The term Q 2.1 = 1 is already displayed in the action field. Change this to Q 2.2 = 1. Position the cursor on this term and press the vertical expand key on the PG (to the right of the segment end key ***). You then obtain an extra input field. Enter A I 2.7 = 1. Press the expand key again and enter the next term etc. The assignments =1 or =0 can be entered with the function keys **F1** and **F2**.

```
PROCESS ELEMENT MONITORING          FB 10/1          CORRECTION

Name : Q 2.2.....                   Station: CARWASH 01 .
Comment:          CARRIAGE AND BRUSH NOT IN START POS. .
Display locations :  00000000       Error ident.:    0....
Movement:      PUL (.) MOT (X)      Reaction mon.: (X)   St cond.: (X)
Release monitoring:                 (F  5.0=1.....                )

Trigger :                           (  .....                      )
Trigger time:          ..... * 100ms

Action :                            (Q  2.2=1.....                )
                                 (A) (I  2.7=1.....               )
                                 (O) (Q  2.1=1.....               )
                                 (A) (I  2.7=1.....               )
Monitoring time:       600 .  * 100ms
Reaction :                          (I  2.3=1.....                )
                                 (A) (Q 2.5=1.....                )
Tolerance t :          1...   * 100ms
Stop condition :                    (Q  2.0=1.....                )

| F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 |
| = 1 | = 0 | = P | = N | DELETE Z | | ENTER | ABORT |
```

Fig. 5-2

When you have filled in the screen form completely, enter your data with **F7**. By pressing **F7** a second time, you can enter the next block.

Now generate the STEP® 5 user program and the setpoint data elements for **carriage forwards** and **carriage backwards** (see also Sections 3.1 and 3.2). These activities are controlled by program block **PB 10**.

Enter the first segment of PB 10. This segment resets the outputs during a cold or warm restart.

PB 10 LEN=35

SEGMENT 1 0000

```
0000  :A   I 2.3    END POSITION CARRIAGE BACK
0001  :AN  Q 2.1    CARRIAGE REVERSE MOTOR
0002  :
0003  :A   I 2.5    END POSITION ROOF BRUSH UP
0004  :R   Q 2.2    ROOF BRUSH UP MOTOR
0005  :AN  I 2.3
0006  :***
```

Once you have pressed the segment end key ***, the PROCESS ELEMENT MONITORING screen form appears. This segment does not, however, require diagnosis, since the function "bring to initial position" is already programmed in FB 10. Simply press **F8 BREAK**, to generate segment 2.

PB 10

SEGMENT 2 0007

```
0007  :A    I  2.7   EMERGENCY STOP
0008  :AN   I  2.0   VEHICLE TOO LONG
0009  :A    I  2.1   VEHICLE POSITION OK
000A  :AN   I  2.2
000B  :A    I  2.3   END POSITION CARRIAGE BACK
000C  :AN   Q  2.1   CARRIAGE  REVERSE  MOTOR
000D  :A    I  3.0   PUSHBUTTON
000E  :S    Q  2.0
000F  :R    F  5.0   START-UP FLAG
0010  :
0011  :A    I  2.4   END POSITION CARRIAGE FRONT
0012  :R    Q  2.0   CARRIAGE  FORWARDS  MOTOR
0013  :***
```

When you complete the segment, the PROCESS ELEMENT MONITORING screen form will once again appear on the screen. Press **F5 CORR**, to allow you to complete the screen form. The cursor is then positioned on the first input field. Complete the input fields as follows:

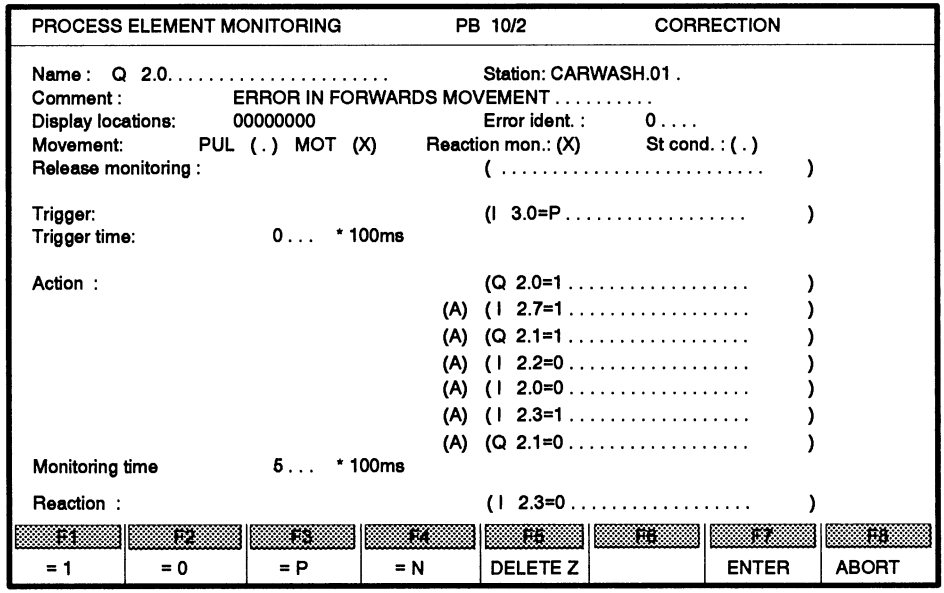


Fig. 5-3

Using the keys **scroll up or down** on the PG, you can move the visible area on the screen. When you have scrolled this area, enter the following:

Tolerance time: **1 * 100ms**
Stop condition: **Q 2.1=1**

After the last input, press **F7 ENTER**. You are now in the output mode. You can now request a second empty screen form for the second setpoint data element by pressing **F3 INSERT**. Then press **F5 CORR**, which allows you to enter the second setpoint data element, as follows:

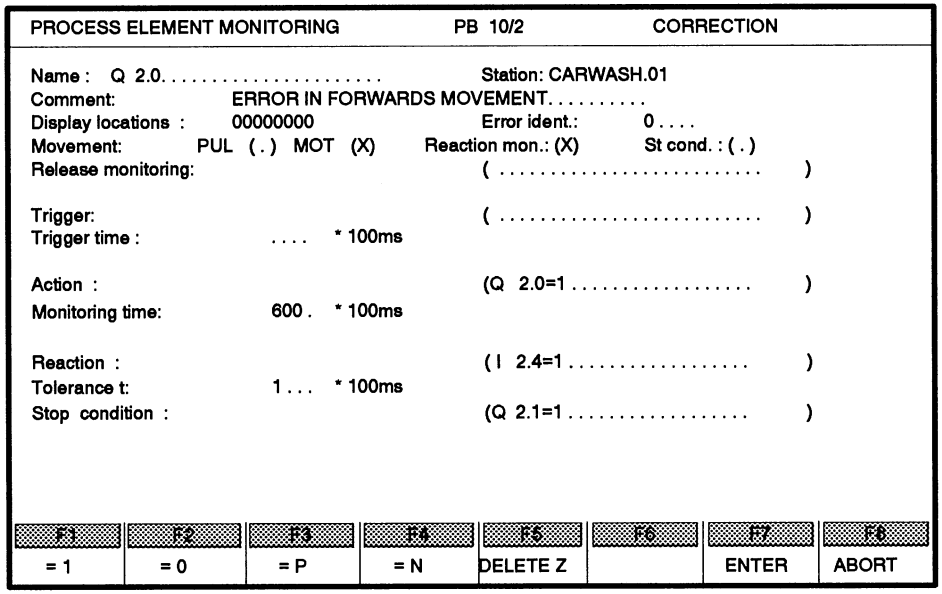


Fig. 5-4

After the last input, press **F7**. The next step is to generate a setpoint data element to monitor the static statuses (see Section 3.1.3). Press **F3 INSERT** and **F4 STAT/DYN**. The screen form MONITORING STATIC SIGNALS is displayed. Press **F5 CORR** to make the inputs. Remember that you must define the **illegal** statuses for status monitoring, as follows:

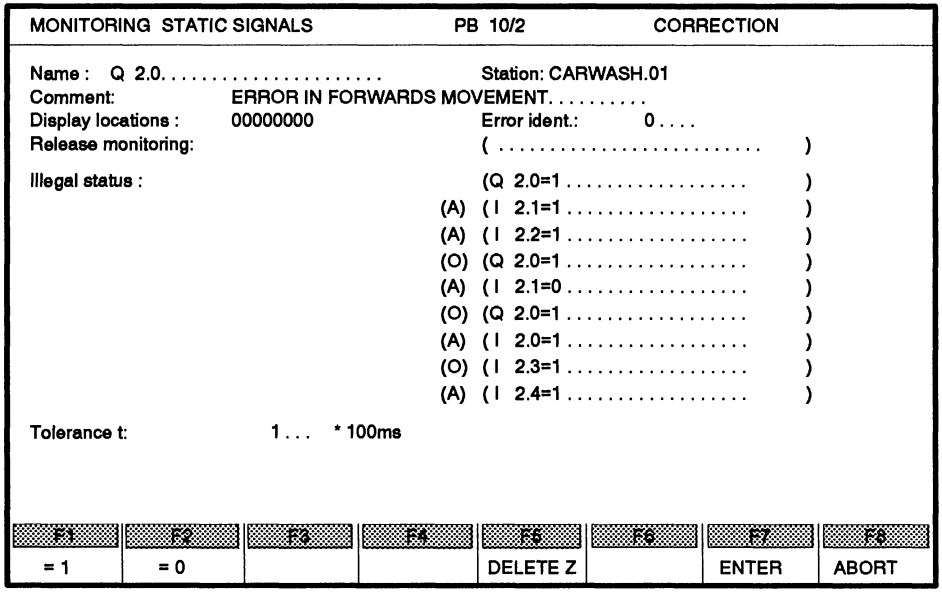


Fig. 5-5

Enter the data with **F7**. Press **F7** again. You can now enter the third segment.

PB 10 LEN=16

SEGMENT 3 0014

```
0014  :A   I 2.7   EMERGENCY STOP
0015  :AN  I 2.0   VEHICLE TOO LONG
0016  :A   I 2.1   VEHICLE POSITION OK
0017  :AN  I 2.2
0018  :A   I 2.4   END POSITION CARRIAGE FRONT
0019  :AN  Q 2.0   CARRIAGE FORWARDS MOTOR
001A  :S   Q 2.1   CARRIAGE REVERSE MOTOR
001B  :
001C  :A   I 2.3   END POSITION CARRIAGE BACK
001D  :R   Q 2.1
001E  :BE
```

Once you have entered the "BE" command and have entered the input, an empty screen form for PROCESS ELEMENT MONITORING is automatically displayed. Make the following entries:

```
PROCESS ELEMENT MONITORING          PB 10/3          CORRECTION

Name : Q 2.1.....                   Station : CAR WASH.01
Comment:          ERROR IN BACKWARDS MOVEMENT .....
Display locations:  00000000        Error ident.:    0....
Movement :    PUL (.) MOT (X)       Reaction mon. : (X)    St cond : (.)
Release monitoring:                 ( .....                )

Trigger:                            (I 2.4=1.....          )
Trigger time:          0...  * 100ms

Action :                            (Q 2.1=1.....          )
                                (U) (I 2.7=1.....          )
                                (U) (I 2.1=1.....          )
                                (U) (I 2.2=0.....          )
                                (U) (I 2.0=0.....          )
                                (U) (Q 2.0=0.....          )
Monitoring time :      5...  * 100ms

Reaction :                          (I 2.4=0.....          )
Tolerance t:           1...  * 100ms

| F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 |
| = 1 | = 0 | = P | = N | DELETE Z | | ENTER | ABORT |
```

Fig. 5-6

Scroll the visible area of the screen and enter the following:

Stop condition: **I 2.3=1**

Enter your input and then generate a further setpoint data element for segment 3 (press **F3 INSERT** in the output mode).

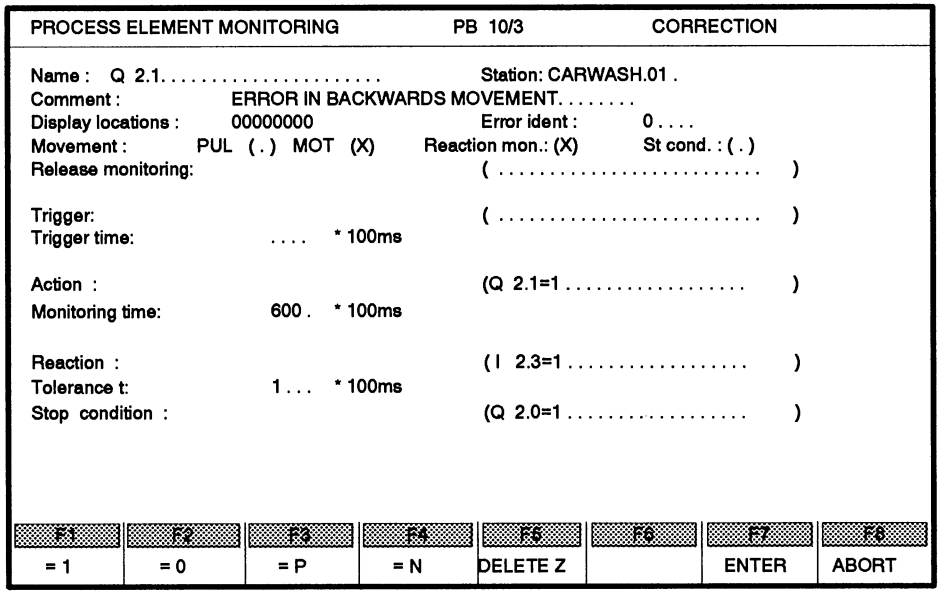


Fig. 5-7

Now generate the setpoint data element for monitoring static statuses (press **F3 INSERT** and **F3 STAT/DYN** in the output mode).

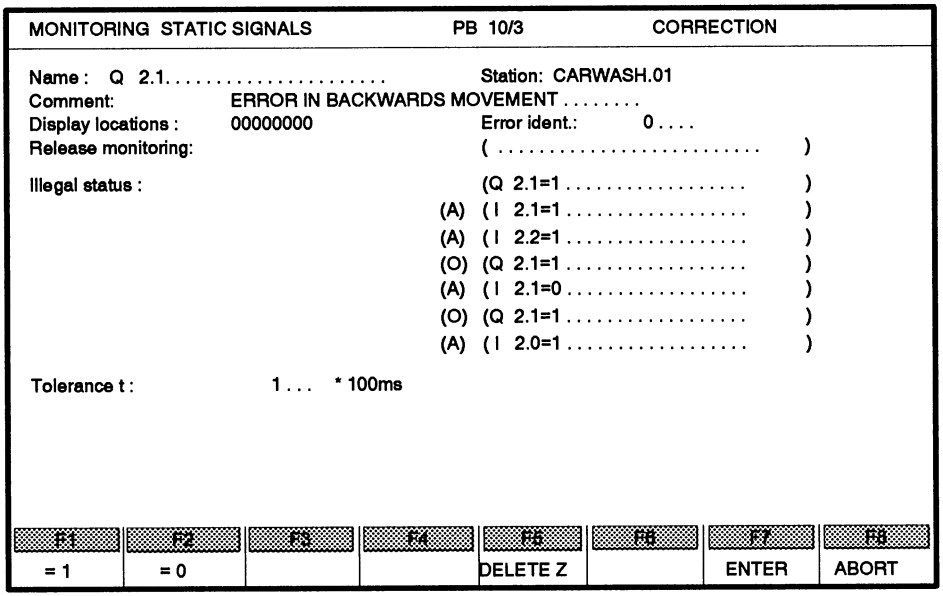


Fig. 5-8

Press **F7** twice to enter the next block.

Now generate the organization blocks **OB 1, OB 20, OB 21 and OB 22** as follows:

```
OB 1                    LEN=17

SEGMENT 1               0000

0000          :A        F 5.0
0001          :JC       FB 10
0002 NAME     :START
0003          :
0004          :JU       PB 10
0005          :
0006          :JU       FB 43
0007 NAME     :KOMCP552
0008 BLCK     :         KF +210
0009 SAMM     :         KF +208
000A          :
000B          :BE
```

Once you have entered "BE" and completed your input, the PROCESS ELEMENT MONITORING screen form is displayed. Break with **F8**.

Generate **OB 20, OB 21 and OB 22** as shown below:

```
OB 20                LEN=17

SEGMENT 1            0000

0000          :O     F 10.0
0001          :ON    F      10.0
0002          :=     F 5.0  SET START-UP FLAG
0003          :
0004          :JU    FB 44
0005 NAME     :DPSTART
0006 SSDP     :      KF +4  Interface no. CP 552
0007 ANZZ     :      KS  N  Group diagnosis via SINEC H1
0008 SSCZ     :      KF +0  Interface no.
                            SINEC CP (CP 535)
0009 BLGR     :      KF +4  Frame size for transfer
                            via SINEC H1
000A DBEA     :      KF +211 DB-EA
000B DBPU     :      KF +209 DB BUFFER
000C MELD     :      KS N   Display of process control messages
000D ANZL     :      KS N   Display also via CP 527
000E SSCL     :      KF +0  Interface no. CP 527
000F          :BE
```

You must now set up the data blocks which were specified in the FB calls (FB DPSTART). These are **DB BLOCK** (e.g. DB 210) and **DB E/A** (e.g. DB 211). For the S5-135U, you also require **DB SAMMEL** (e.g. DB 208). Set up the data blocks **DB BLOCK** and **DB SAMMEL** with a length of 261 words. The **DB E/A** data block should be 276 words in length.

You must now transfer the following blocks to the PLC:

OB 1, OB 20, OB 21, OB 22,
DB 210 (DB BLOCK), DB 211 (DB E/A), and DB 208 (DB SAMMEL)
FB 10 and PB 10
FB 43 and FB 44 (called in OB 1 and OB 20 / 21 / 22)
DB 7 (used by FB 43 and FB 44)

To transfer the setpoint data to the CP 552, call the programming package **COM 552**. Complete the PRESETS screen form as follows: (you can make your entries with F3 SELECT).

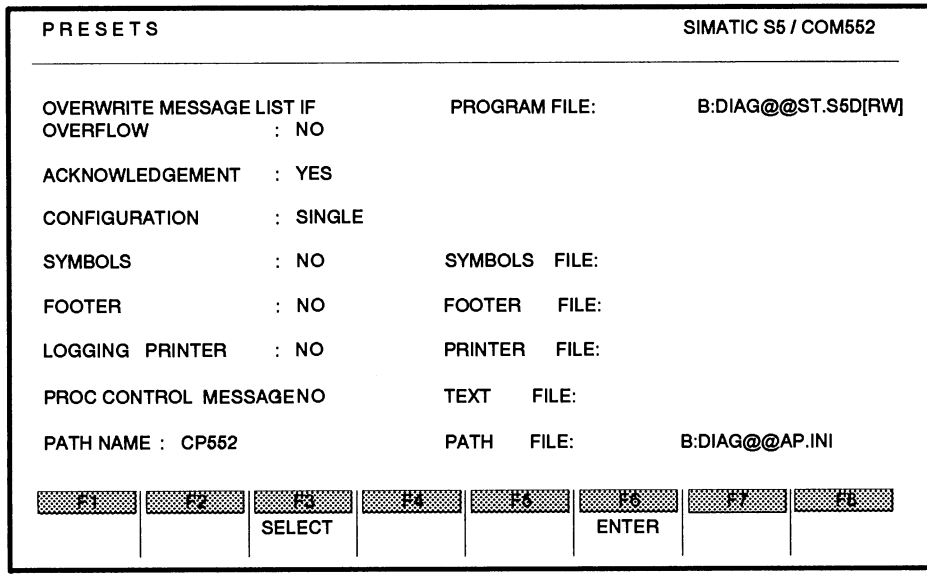
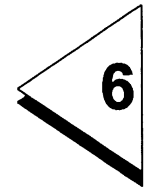


Fig. 5-9

When you enter the input with **F6**, the screen form SELECT FUNCTION is displayed. Press **F7 SCREEN FORM EDITOR** and then **F3 TRANSFER**. If you press **F3 again**, the setpoint data is transferred from the hard disk to the CP 552. Enter

B

in the SP DATA FOR BLOCK field to transfer the setpoint data for all blocks and then press the enter key. Return to function selection by pressing **F8** twice.



Error Displays

Contents

You plan the error display on the programmer and on the local monitor using the programming package COM 552.

First, set the SYSID of the CP 552 module. In the SELECT FUNCTION screen form press **F4 SPECIAL FUNCTION** and **F3 EDIT SYSID**. The SYSID information is then displayed. For PLANT, enter the plant designation; in this example this should be as follows:

DIAGPROCESSOR01

Make sure that this designation is identical to the path name you specified in Section 4.2 to select the CPU automatically.

In the DATE field, you can enter the date on which changes are made. This date does not influence the date set by the hardware clock of the CP 552.

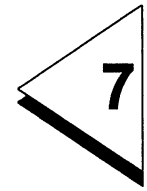
In single diagnosis, the fields DISPLAY LOCATIONS FOR SYSTEM MESSAGES/PROCESS CONTROL MESSAGES should be 0.

If you require error display on the local monitor you must specify CONNECT LOCAL MONITOR: "YES." For further information about the local monitor, refer to the User's Guide "Error Display on the Local Monitor."

Then press **F7 DONE** and **F8 RETURN** twice. You are now once again in the SELECT FUNCTION screen form.

If you press **F3 CP 552 ONLINE** in the SELECT FUNCTION screen form, you can set the date and time of the hardware clock of the CP 552. Press **F4 DATE/TIME**. Then press **F1** and enter the date (format dd:mm:yy) and enter this with **F6**. You can set the time with **F2** (format hh:mm:ss). Enter the time with the RETURN key. If you now press **F6** you start the clock. If you then press **F8** twice, you return to the SELECT FUNCTION screen form.

The error display is started on the programmer by pressing **F1 CURRENT MESSAGES** in COM 552. You now only need to start the CP 552 and the CPU. You first receive the system message "CP 552 START-UP." Acknowledge this message with **F1**. Following this, an asterisk appears on the screen. Using the simulator, you can now simulate the process and generate process errors (e.g. I 2.3 = 1 and I 2.4 = 1 as an illegal status), this will then be displayed on the screen.



Group Diagnosis via SINEC H1

Contents

This section of the example explains how to carry out group diagnosis via the SINEC H1 bus system. Before trying out the following steps, you should already have carried out the steps described in Sections 1 to 6.

In addition to the hardware required so far (see Section 4), you also require the following for group diagnosis:

- a SINEC H1 bus system
- two programmers (PG 685, PG 750), each with a SINEC-H1 CP (CP 536 or CP 141)
- one SINEC-H1 CP (CP 535 or CP 143)

7.1 General Notes

The system configuration is as follows:

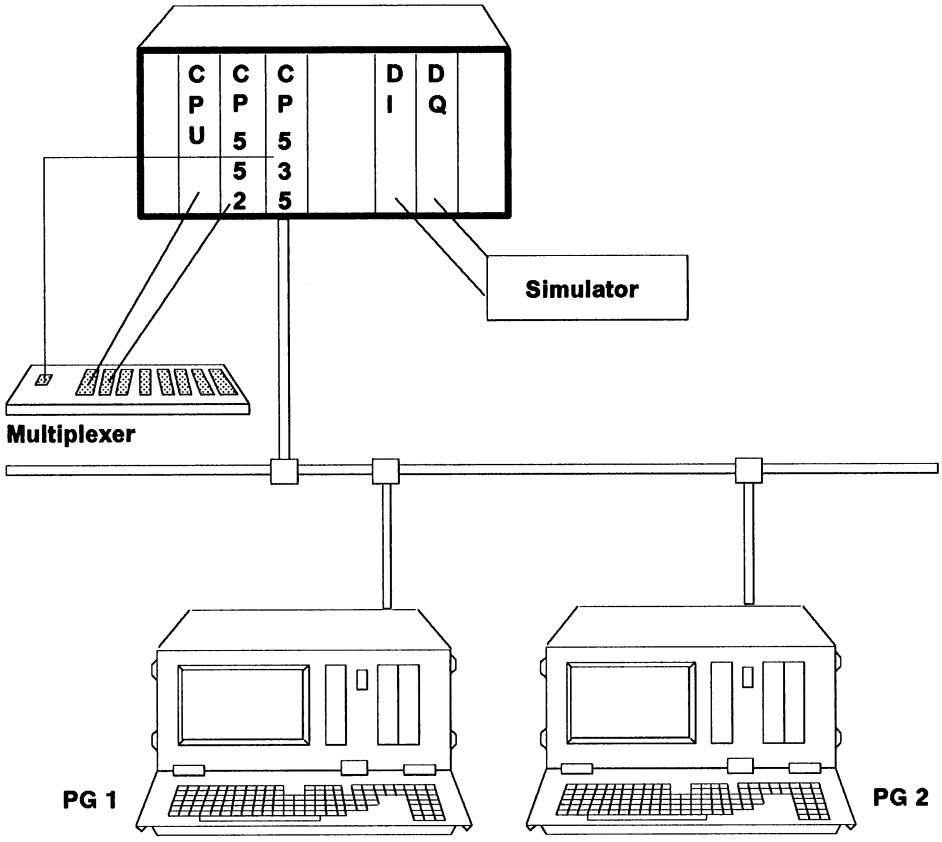


Fig. 7-1 System configuration for group diagnosis

The error messages for **carriage forwards** will be displayed on display unit **PG 1**, the error messages for **carriage backwards** on **PG 2** and the error messages for **start-up** (FB 10) on both display units.

The error messages are sent by the CP 552 to the CPU, from the CPU to the CP 535 and from the CP 535 via the SINEC H1 bus to the CP 536 in the display unit.

Set up the system as explained in the installation guidelines. For the CP 535, set the interface number (SSNR) 4. The CP 552 should retain interface number 0.

The application example has been generated using a PG 685 with CP 536 and one CP 535 (parameters assigned with COM 535) in the PLC.

7.2 Stipulating Display Locations for Error Messages

When using group diagnosis, you must stipulate in each setpoint data element the display units on which the corresponding error message will be displayed. In the diagnostic screen form, there is therefore a field for **display locations**. An error message can be assigned to a maximum of 8 display units.

The procedure is as follows: first establish the cable connection from the display unit to the CP 552 and call the programming package COM 552. Enter the name

B:DIAG@@ST.S5D

as the **program file** in the PRESETS screen form and enter this name with **F6**. Change to the SCREEN FORM EDITOR with **F7**. Press **F2 OUTP/SP**. Output the setpoint data elements to **FD** for **PB 10, segment 2**. The following setpoint data element appears on the screen:

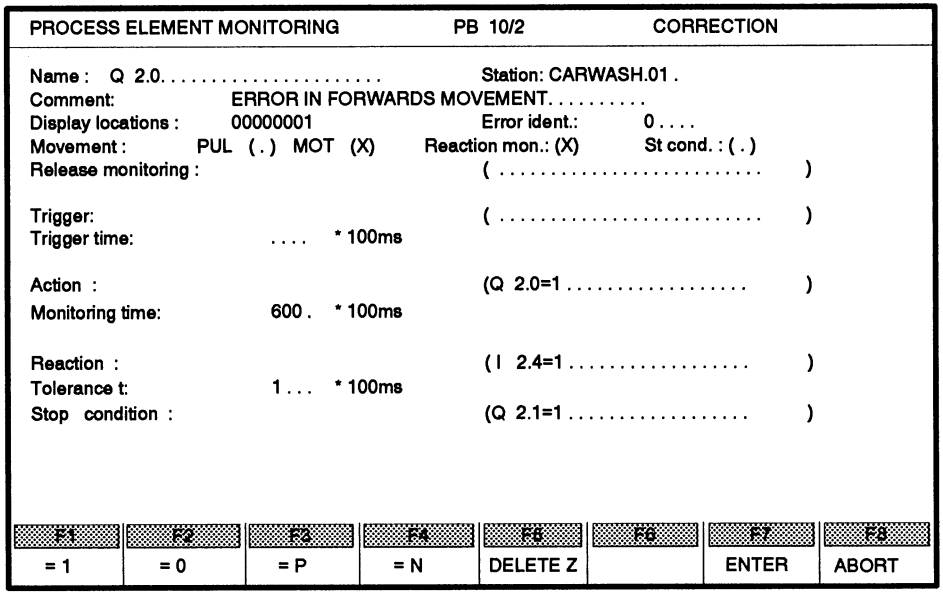


Fig. 7-2

Press **F5 CORR**, to make changes. Move the cursor to the display locations field. Since the error messages for **carriage forwards** are to be displayed on **PG 1**, replace the right-hand "0" with a "1."

Display locations: **00000001**

Enter with **F7**. Select the next setpoint data element by pressing **F2 ELEMENT + 1**. Once again change the display location:

Display locations: **00000001**

Repeat this with the third setpoint data element for segment 2. Finally, press **F7** twice. The same procedure is required for the setpoint data elements for segment 3.

The error messages for **carriage backwards** are to be displayed on **PG 2**; enter the following:

Display locations: **00000010**

Follow the same procedure for the setpoint data element for **FB 10**. Since the error messages for the start-up are to be displayed on both programmers, enter the following:

Display locations: **00000011**

Finally, transfer the setpoint data elements for **PB 10** and **FB 10** from the hard disk to the CP 552. In the SCREEN FORM EDITOR screen form press **F3 TRANSFER** and then **F3** again. In the field "SP data for block" enter "A." Then press the enter key. The setpoint data elements for all blocks (here, PB 10 and FB 10) are transferred.

7.3 Assigning Parameters to the CP 552

In the COM 552 programming package you stipulate the display locations for system messages. Press **F3 CP 552 ONLINE** and **F7 SYSID**. Position the cursor and enter the following:

DISPLAY LOCATIONS FOR SYSTEM MESSAGES: **00000001**

This means that the system messages are output on display unit 1 (= PG 1). Enter the plant designation in the PLANT field as follows:

PLANT: **DIAGPROCESSOR01**.

This plant designation must also be specified when you generate the configuration file for the display units. Transfer this information with **F7** and then exit the programming package with **F8**.

7.4 Assigning Parameters to the CP 535

The next task is to determine the physical and logical station addresses. Keep the following data flow in mind:

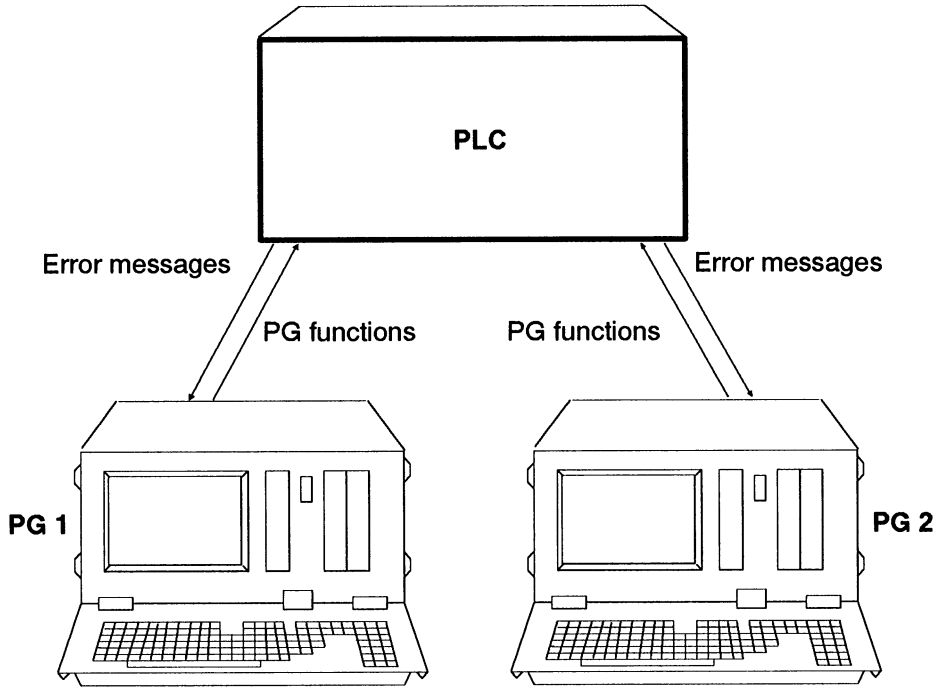


Fig. 7-3 Data flow

This data flow is only possible when the programmable controller knows the display unit and when the display units know the programmable controller.

To establish the links, the Ethernet addresses, TSAP-IDs and job numbers must be assigned:

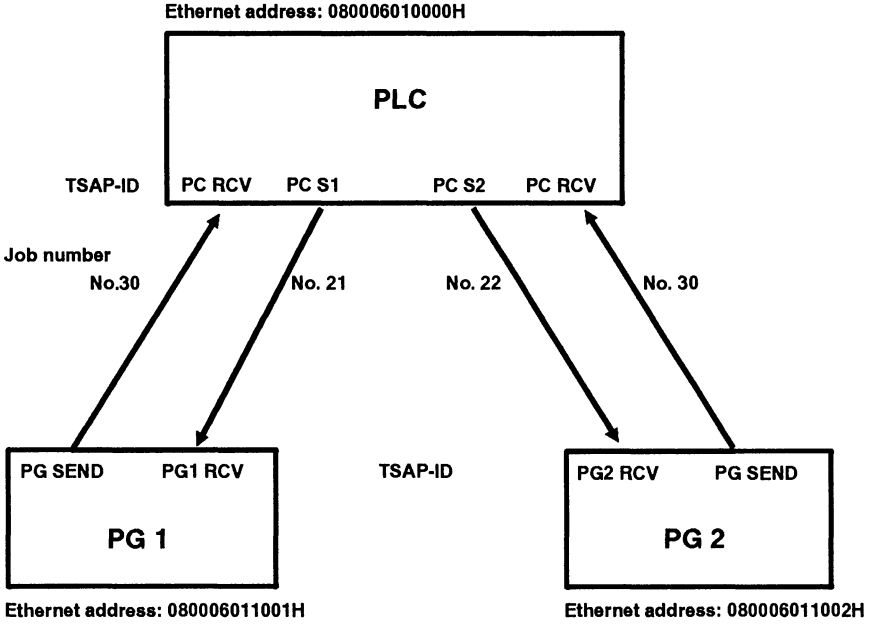


Fig. 7-4 Assigning parameters to the CP 535

Both display units transmit to the PLC. The **CP 535** in the PLC must have the following **RECEIVE job** available to allow it to receive.

- **RECEIVE job** Job number e.g. 30

  own TSAP-ID (CP 535, receive) e.g. PC_RCV
  remote TSAP-ID (CP 536, send) e.g. PG_SEND

The programmable controller sends error messages to both display units. **Two SEND jobs** must therefore be prepared in the PLC as follows:

- **SEND job 1** job number e.g. 21

  own TSAP-ID (CP 535, send) e.g. PC_S1
  remote TSAP-ID (CP 536, receive) e.g. PG1_RCV

- **SEND job 2** job number e.g. 22

  own TSAP-ID (CP 535, send) e.g. PC_S2
  remote TSAP-ID (CP 536, receive) e.g. PG2_RCV

The **Ethernet addresses** are assigned as follows:

- Prog. controller: 080006010000H
- Programmer PG 1: 080006011001H
- Programmer PG 2: 080006011002H

You assign parameters to the CP 535 with the programming package COM 535. Detailed information about this can be found in the manual for the CP 535/COM 535.

In the programming package COM 535, you must complete or check the screen forms SYSTEM IDENTIFICATION and INITIALIZATION BLOCK (see Figs. 7-5 and 7-6). First of all, connect the programmer to the appropriate interface of the CP 535. Remember that when you are working ONLINE you can only change the SYSID and INIT block when the CP 535 is set to STOP. Transfer the blocks to the CP 535 by switching the RUN/STOP switch of the CP 535 from STOP to RUN.

SYSTEM IDENTIFICATION — SIMATIC S5 / COM535

CP TYPE	: CP 535	VERSION	: V 4.2		
PASSWORD	:	DATE	:		
MUX-ADR	:	SLAVE NO	:	BASE SSNR	: 4
MOD TYPE	: RAM	MOD LENGTH	: 32 KBYTES		
ETHERNET ADDR	: 080006010000 H				
NSAP-ID	:				

F1	F2	F3	F4	F5	F6	F7	F8
						DONE	RETURN

Fig. 7-5

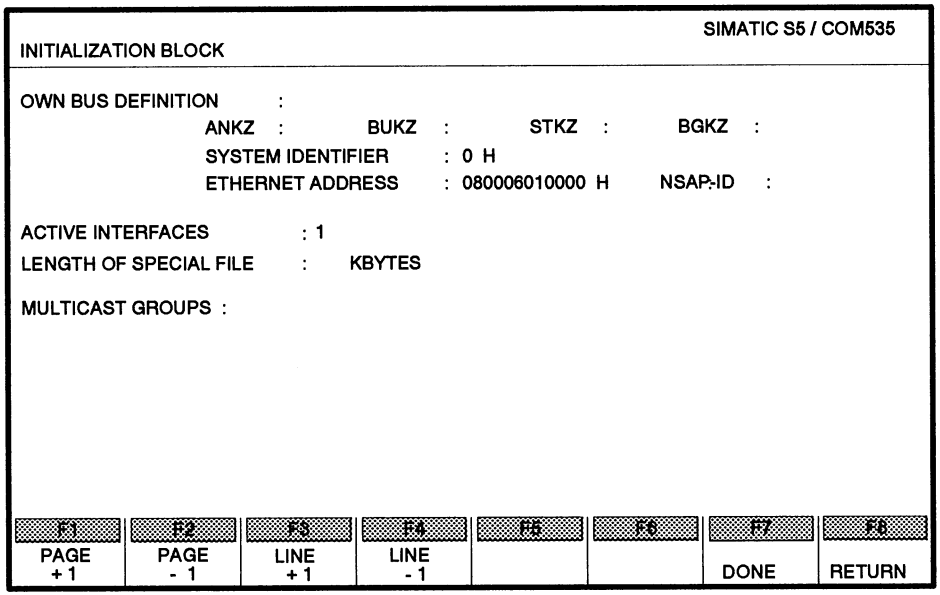


Fig. 7-6

Following this, you must establish link blocks for the RECEIVE job and the SEND jobs (see Figs. 7-7 and 7-8).

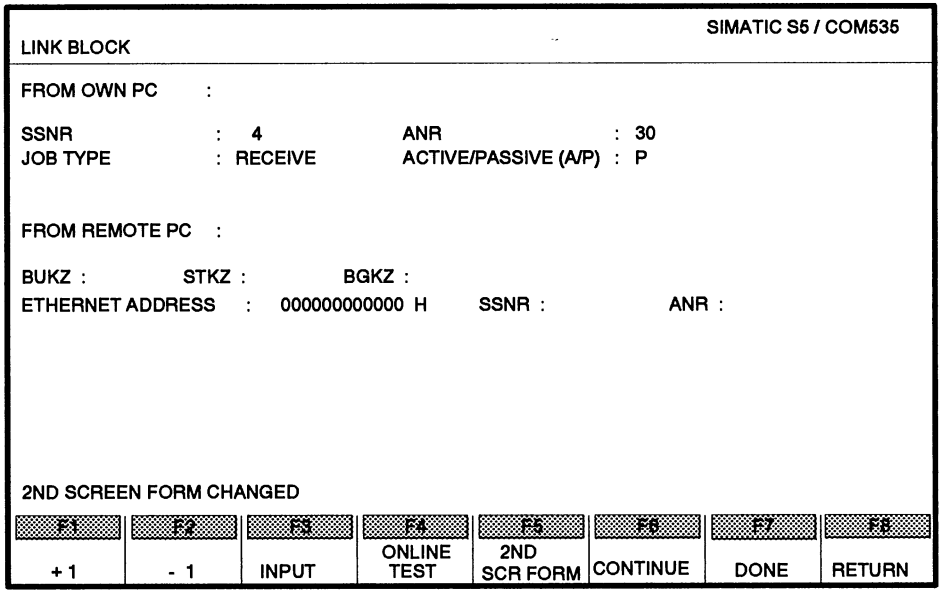

SIMATIC S5 / COM535
LINK BLOCK
FROM OWN PC :
SSNR : 4
ANR : 30
JOB TYPE : RECEIVE
ACTIVE/PASSIVE (A/P) : P
FROM REMOTE PC :
BUKZ : STKZ : BGKZ :
ETHERNET ADDRESS : 000000000000 H
SSNR :
ANR :
2ND SCREEN FORM CHANGED
F1 + 1
F2 - 1
F3 INPUT
F4 ONLINE TEST
F5 2ND SCR FORM
F6 CONTINUE
F7 DONE
F8 RETURN

Fig. 7-7

```
LINK BLOCK                                          SIMATIC S5 / COM535
                                                    2ND SCREEN FORM

MULTICAST (Y/N)    : N   MULTICAST GROUP:      ETHERNET ADDRESS :        H

DATAGRAM (Y/N)     : N
PRIORITY           : 4   READ/WRITE (Y/N) : N

SOURCE/DEST        :                  LENGTH   :
COND CODEWORD      :
INTERPRETER        :                  ADDRESS  :        :        H

OWN TSAP-ID        : LENGTH :  6   HEX :  41 47 20 52 43 56 20     ASC :  AG RCV
REMOTE TSAP-ID     : LENGTH :  7   HEX :  50 47 20 53 45 4E 44     ASC :  PG SEND
REMOTE TSAP-ID     : LENGTH :  12  ASC :

NUMBER OF JOBS PER TSAP :  1

| F1  | F2  | F3 | F4 | F5          | F6   | F7   | F8     |
|     |     |    |    | 1ST         |      |      |        |
| + 1 | - 1 |    |    | SCR FORM    | HELP | DONE | RETURN |
```

Fig. 7-8

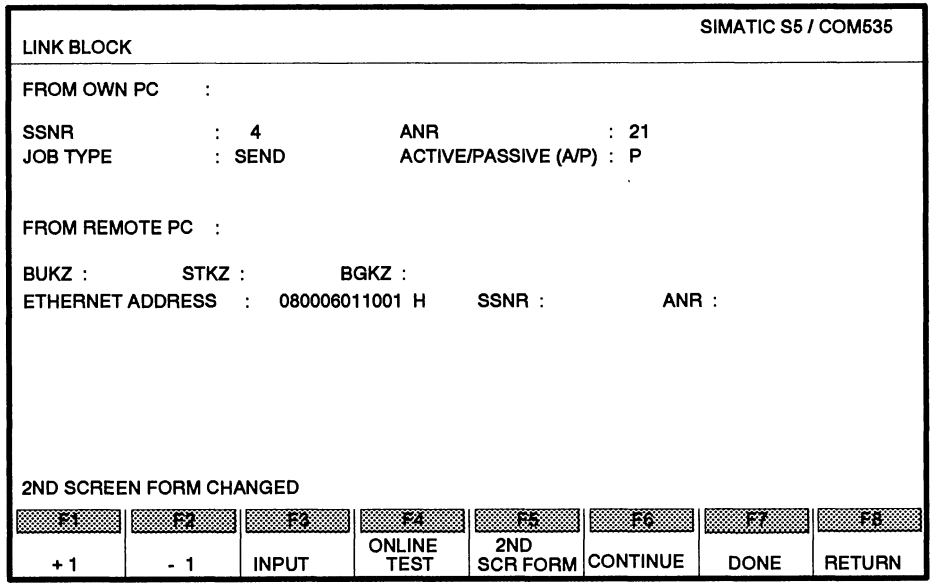

SIMATIC S5 / COM535
LINK BLOCK
FROM OWN PC :
SSNR : 4
ANR : 21
JOB TYPE : SEND
ACTIVE/PASSIVE (A/P) : P
FROM REMOTE PC :
BUKZ : STKZ : BGKZ :
ETHERNET ADDRESS : 080006011001 H
SSNR :
ANR :
2ND SCREEN FORM CHANGED
F1 +1
F2 - 1
F3 INPUT
F4 ONLINE TEST
F5 2ND SCR FORM
F6 CONTINUE
F7 DONE
F8 RETURN

Fig. 7-9

```
LINK BLOCK                                          SIMATIC S5 / COM535
                                                    2ND SCREEN FORM

MULTICAST (Y/N)   : N   MULTICAST GROUP:     ETHERNET ADDRESS :      H

DATAGRAM (Y/N)    : N
PRIORITY          : 4   READ/WRITE (Y/N) : N

SOURCE/DEST       :                  LENGTH   :
COND CODEWORD     :
INTERPRETER       :                  ADDRESS  :      :      H

OWN TSAP-ID       : LENGTH :         HEX  :41 47 20 53 31 20 20     ASC :  PC S1
REMOTE TSAP-ID    : LENGTH :  7      HEX  : 50 47 31 20 52 43 56    ASC:   PG1RCV
REMOTE TSAP-ID    : LENGTH :  12     ASC  :

NUMBER OF JOBS PER TSAP :  1

| F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 |
|----|----|----|----|----|----|----|----|
| +1 | - 1 |  |  | 1ST SCR FORM | HELP | DONE | RETURN |
```

Fig. 7-10

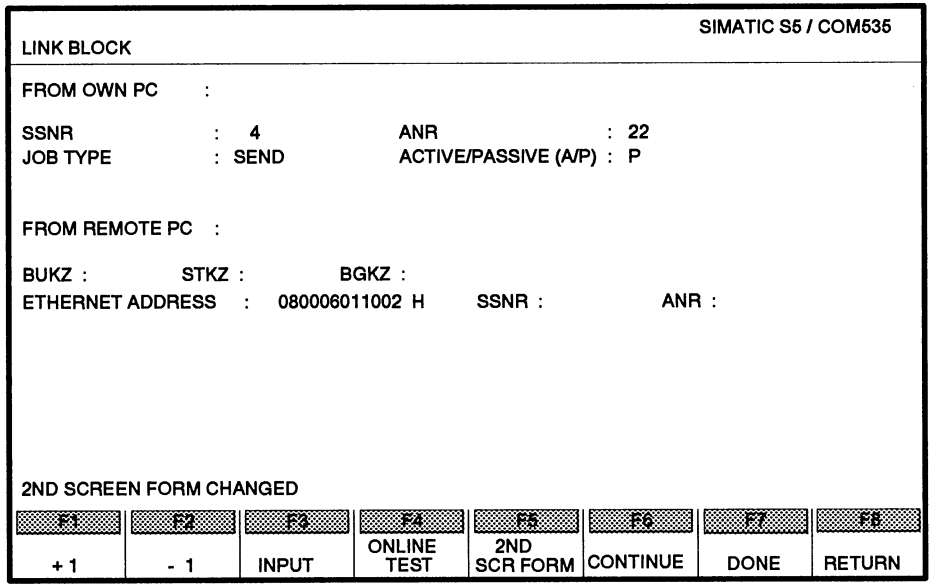

SIMATIC S5 / COM535
LINK BLOCK
FROM OWN PC :
SSNR : 4
ANR : 22
JOB TYPE : SEND
ACTIVE/PASSIVE (A/P) : P
FROM REMOTE PC :
BUKZ : STKZ : BGKZ :
ETHERNET ADDRESS : 080006011002 H
SSNR : ANR :
2ND SCREEN FORM CHANGED
F1 +1
F2 - 1
F3 INPUT
F4 ONLINE TEST
F5 2ND SCR FORM
F6 CONTINUE
F7 DONE
F8 RETURN

Fig. 7-11

```
LINK BLOCK                                          SIMATIC S5 / COM535
                                                    2ND SCREEN FORM

MULTICAST (Y/N)   : N   MULTICAST GROUP:     ETHERNET ADDRESS :        H

DATAGRAM (Y/N)    : N
PRIORITY          : 4   READ/WRITE (Y/N) : N

SOURCE/DEST       :                  LENGTH   :
COND CODEWORD     :
INTERPRETER       :                  ADDRESS  :        :        H

OWN TSAP-ID       : LENGTH :        HEX :  41 47 20 53 32  20 20   ASC :  PC S2
REMOTE TSAP-ID    : LENGTH :  7     HEX :  50 47 32 20 52  43 56   ASC :  PG2RCV
REMOTE TSAP-ID    : LENGTH :  12    ASC :

NUMBER OF JOBS PER TSAP :  1

| F1  | F2  | F3 | F4 | F5          | F6   | F7   | F8     |
|-----|-----|----|----|-------------|------|------|--------|
| +1  | - 1 |    |    | 1ST SCR FORM| HELP | DONE | RETURN |
```

Fig. 7-12

7.5 Assigning Parameters to the CP 536

Now assign parameters to the two CP 536 processors in PG 1 and PG 2. Begin with the CP 536 in PG 1:

In the SELECT PACKAGE screen form, select the CP 536 interface with **F5 INTERFACE**. Then press **F2 UTILITY** followed by **F1 BUS SEL**. Enter a PATH NAME (e.g. "CP 536") and a PATH FILE (e.g. "CP536PAP.INI). With **F1 EDIT** you call the editing mode. Press **F3 CP 536**. Enter the settings with **F6**. Now activate the path to the CP 536 interface with **F2 ACTIVE** and **F3 SINGLE** (see Fig. 7-12). You obtain the message "PG DIRECT LINK ESTABLISHED". At the upper right-hand edge of the screen, an "asterisk" will appear in the "ACTIVE" column.

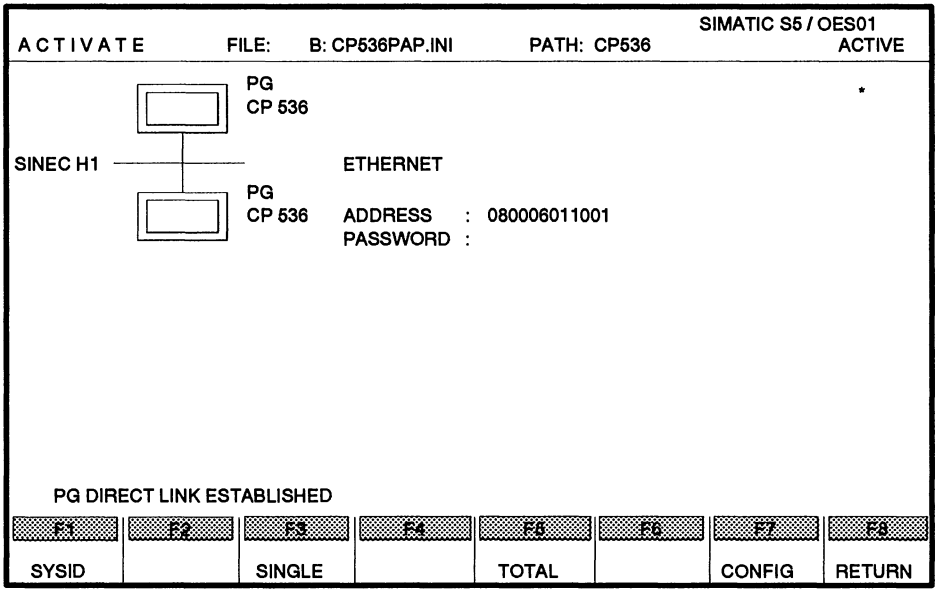


Fig. 7-13

Call the system identification block of the CP 536 by pressing **F1 SYSID**. Enter the Ethernet address assigned to the PG as follows:

	PG 1	**PG 2**
Ethernet address	**080006011001 H**	**080006011002 H**

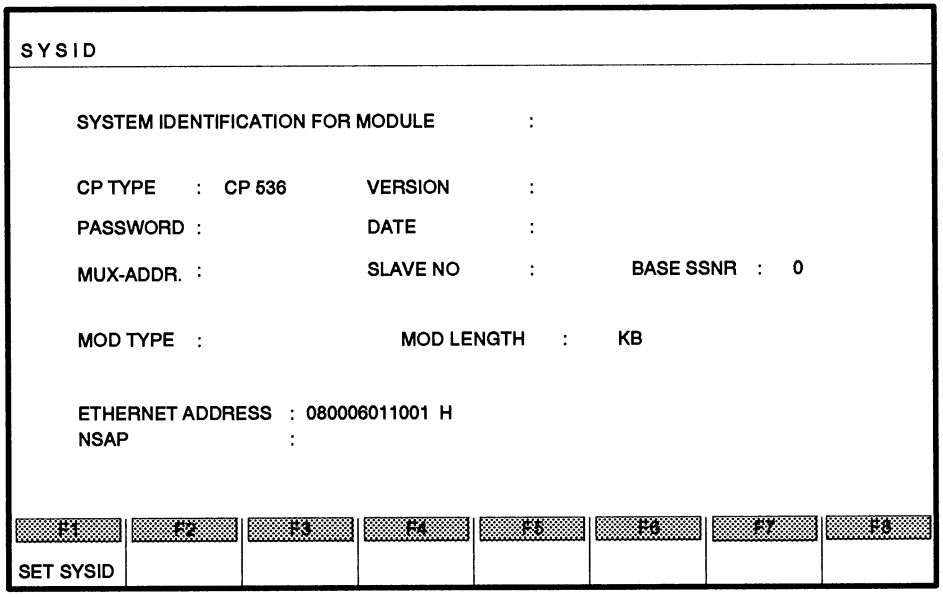


Fig. 7-14

Set the SYSID with the function key **F1 SET SYSID**. You have now assigned parameters to the CP 536 interface.

Repeat the procedure for the CP 536 in **PG 2**. Here, use the Ethernet address 080006010002.

7.6 Generating the Configuration File

You must generate a configuration file for each connected display unit. In the configuration file you specify the diagnostic processors with which the display unit will communicate.

First, generate the **configuration file for PG 1**. Select the interface **CP 536** in the SELECT PACKAGE screen form with **F5** and call the programming package **COM 552**. Enter the preset with **F6** and press **F4 SPECIAL FUNCTION**. If you now press **F1 EDIT CONFIG** the configuration editor is activated. enter the following name for the configuration file:

B:PG1CONKF.INI

Enter this name with **F6**. Since you have specified a file name that does not yet exist, the screen form NEW PATH is displayed.

You must complete a screen form for each programmable controller with which the display unit is to communicate. In this example, you only need to complete one screen form, since only one PLC is connected.

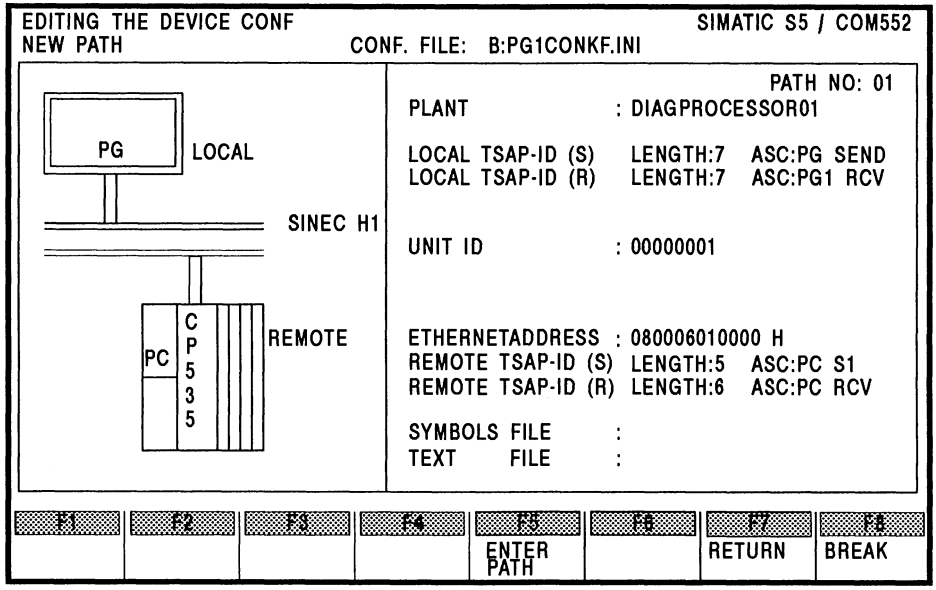


Fig. 7-15

Note:

The plant designation in the PLANT field must be identical to the one in the **SYSID** of the CP 552.

Enter the input with **F5**. Then press **F7** twice followed by **F8**.

You must also generate a configuration file for **PG 2**. Here, use the file name

B:PG2CONKF.INI

and enter the following:

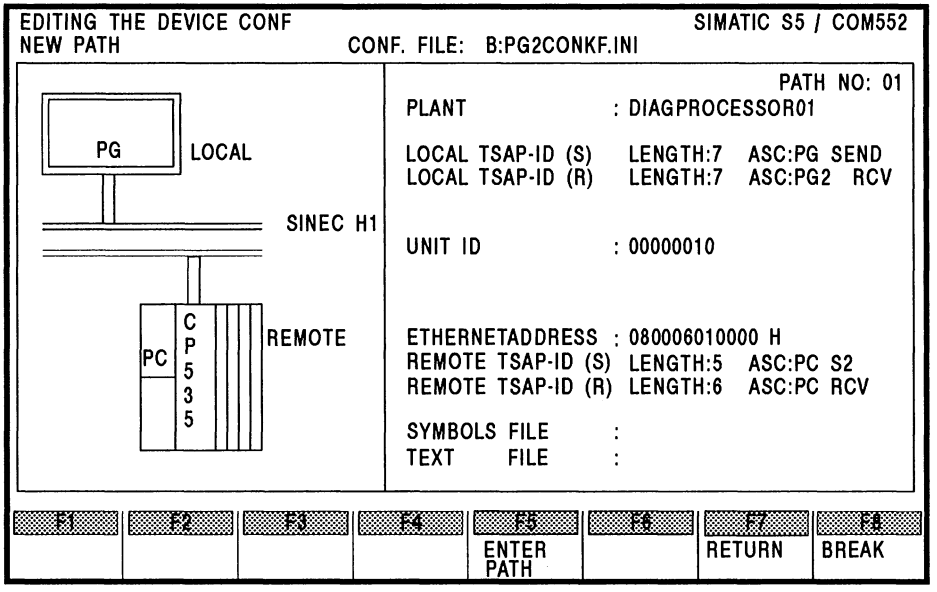


Fig. 7-16

7.7 Generating a Path File

To be able to display the segment status, BSTACK and ISTACK of the PLC via the PG channel of the CP 535 when errors are signalled, you must generate a path file on each display unit. Once again, start with **PG 1**.

Press **F2 UTILITY** in the S5 screen form SELECT PACKAGE. Press **F1** to activate the **BUS SELECTION**. Now enter the following:

PATH NAME: **PC**
PATH FILE: **B:P1PATHAP.INI**

Press **F1** to call the editor with which you can generate a path.

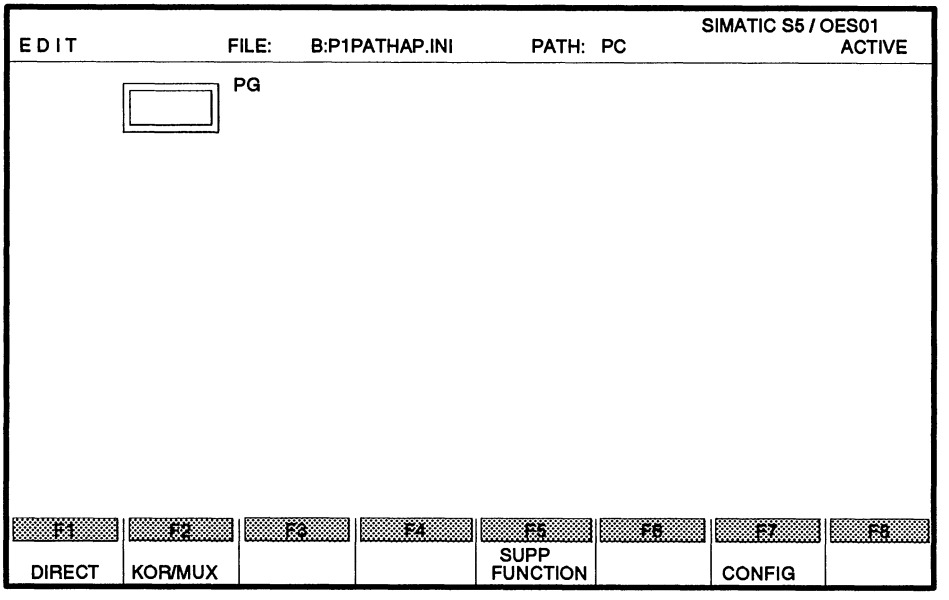


Fig. 7-17

In the EDIT screen form press **F1 DIRECT** and then press **F2 CP 535** twice. Enter the following:

ETHERNET ADDRESS: **080006010000**

Now press **F2 KOR/MUX** and then **F1 ENDP**. This path should have the following end point address:

1

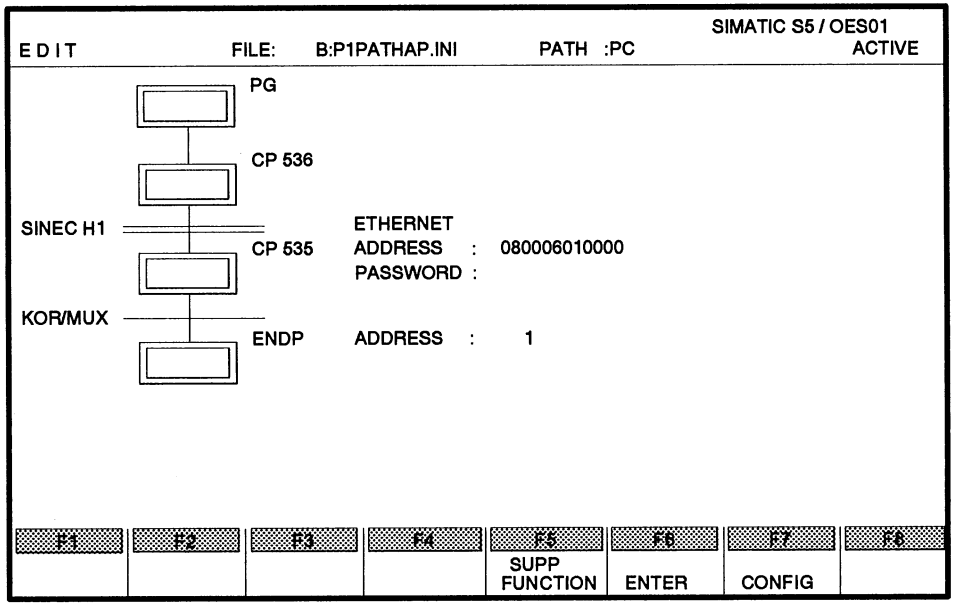


Fig. 7-18

Press **F6 ENTER** to enter your data. The basic screen form for BUS SELECTION is then displayed.

Following the same procedure, generate two further paths with the following names and addresses:

PATH NAME	ETHERNET ADDRESS	ENDP ADDRESS
DIAGPROCESSOR01	**080006010000**	**1**
CP552	**080006010000**	**2**

Now repeat this procedure for **PG 2**. Here, use the following name for the path file:

B:P2PATHAP.INI.

7.8 STEP® 5 User Program for Group Diagnosis

To display errors via the SINEC H1 bus system, you require further function blocks and another data block in your STEP® 5 user program. The function blocks **FB 45** and **FB 46** must be called in organization block **OB 1**. **FB 47** does not need to be called explicitly but must exist in the PLC, since it is called by FB 46.

You have set up three link blocks for the CP 535 (see Section 7.4) for data exchange via the SINEC H1 bus as follows:

RECEIVE job with A-NR = 30	PLC receives from PG 1 and PG 2
SEND job with A-NR = 21	PLC sends to PG 1
SEND job with A-NR = 22	PLC sends to PG 2

To exchange data between the CP 535 and the CPU you must assign the following parameters for **FB 45** and **FB 46**:

OB 1

```
      : JU  FB 45
NAME  : RECEIVE
ANRZ  : KF +30        Job number of the RECEIVE function
                      of the SINEC-H1-CPs
ANRL  : KF +0         Job number of the RECEIVE function
                      of the CP 527

      : JU  FB 46
NAME  : SEND
ANZ1  : KY 21,0       Job number for the send job to the display unit,
                      right byte: irrelevant.
                      Setpoint data elements with the display
                      location: xxxx xxx1 (x signifies unimportant (0 or 1) )
                      are sent with this A-NR
ANZ2  : KY 22,0       Job number for the send job to the display unit,
                      right byte: irrelevant.
                      Setpoint data elements with the display
                      location: xxxx xx1x are sent with this A-NR.

ANZ3  : KY  0,0
ANZ4  : KY  0,0       Display units
ANZ5  : KY  0,0       3 to 8 are
ANZ6  : KY  0,0       not connected
ANZ7  : KY  0,0
ANZ8  : KY  0,0
```

Now establish data block **DB PU** (e.g. DB 209) as message buffer with a length of **375**.

You must now change the parameters of **FB 44** in the organization blocks **OB 20, 21 and 22** as follows:

NAME : DPSTART	
SSDP : KF +4	Interface no. CP 552
ANZZ : KS J	Group diagnosis via SINEC H1
SSCZ : KF +0	Interface no. SINEC CP (CP 535)
BLGR : KF +4	Frame size for transfer via SINEC H1
DBEA : KF +211	DB-EA
DBPU : KF +209	DB Buffer
MELD : KS N	Display of process control messages
ANZL : KS N	Display also via CP 527
SSCL : KF +0	Interface no. CP 527

Now synchronize the CP 552 and CP 535 by switching the STOP/RUN switch on the CPU.

7.9 Error Displays in COM 552

The error messages are displayed in COM 552 on the programmer and on the local monitor if this is connected. For further information refer to the User's Guide "Error Display on the Local Monitor."

Proceed as follows to display errors on the PG. Select the programming package COM 552 at PG 1 and complete the PRESETS screen form as follows:

PRESETS			SIMATIC S5 / COM552
OVERWRITE MESSAGE LIST IF OVERFLOW	: NO	PROGRAM FILE	: B:DIAG@@ST.S5D [RW]
ACKNOWLEDGEMENT	: YES		
CONFIGURATION	: GROUP	CONFIG FILE	: B:PG1CONKF.INI
SYMBOLS	: NO		
FOOTER	: NO	FOOTER FILE	:
LOGGING PRINTER	: NO	PRINTER FILE	:
PATH NAME : CP552		PATH FILE	: B:P1PATHAP.INI

F1	F2	F3	F4	F5	F6	F7	F8
		SELECT			ENTER		

Fig. 7-19

Enter the information with **F6**. The SELECT FUNCTION screen form is now displayed. To display errors, press **F1 CURRENT MESSAGES**.

Repeat this procedure with **PG 2**. In this case, enter the following in the CONFIG FILE field:

B:PG2CONKF.INI

Both programmers are now ready for error display. Using the simulator, you can now simulate the process and generate process errors, which are then displayed on the two programmers.