SIMATIC S5

COM 95F
Parameterization Software

Manual

This manual is part of the software package
with the order number:

6ES5 895-6MF23

EWA 4NEB 812 6221-02a

Edition 02

EWA 4NES 812 6213-02

Summary

Page

Introduction

Please read the introduction carefully. You will then find it easier to use the manual and this will save time.

COM 95F is parameterization software for the S5-95F programmable controller. COM 95F supports you in system parameterization and, in the event of a fault, it makes error locating and analysis easier. As well as the COM 95F parameterization software, the software package supplied also contains an acceptance tool which makes acceptance of the user program easier.

In this manual we have attempted to present the necessary information as completely and as well organized as possible. Certain information is repeated in various chapters so that you do not have to leaf through the manual to find what you need.

On the following pages of this introduction you will find information that will make it easier for you to use the manual. This section explains how the manual is organized.

Description of contents

- Chapter 1
  This chapter gives an overview of the files required for the COM 95F parameterization software and tells you how to install and start the package on your programmer.

- Chapter 2
  This chapter describes the COM 95F screen forms in which you must enter your system parameters. In addition, the meanings of all the parameters are explained briefly.

- Chapter 3
  This chapter uses examples to show how COM 95F outputs the error messages and static error images of the S5-95F.

- Chapter 4
  This chapter gives an example of a complete system parameterization printout.

- Chapter 5
  This chapter describes the use of the acceptance tool. Simple examples make the first steps easy.

Scope of the manual

This manual contains a detailed description of the COM 95F software package, Versions V 2.1 and V 7.01. The software package is intended for the S5-95F programmable controller with basic unit 6ES5-095-8FB01. You can also use the software package for parameterizing the S5-95F programmable controller with basic units 6ES5 095-8FA01 and 6ES5 095-8FA02. In this case, you must be careful to set only the parameters permissible for these units. The S5-95F programmable controllers are described in separate manuals.

Aim of the manual

This manual describes the installation and operation of the COM 95F parameterization software and the acceptance tool. We assume that you are acquainted with the hardware components of the S5-95F and that you have an S5-95F manual.

To be able to work successfully with this manual, you should have experience of working with STEP 5 and KOMDOK; experience with our COM software packages would help but is not essential.

There are correction forms inserted at the end of the manual. Please enter your suggestions/ corrections here and send the form back to us. This will help us to improve the next edition.

Conventions

This system manual is organized in menu form to make it easier for you to find information. This means the following:

- Each chapter is marked with printed tabs.
- At the front of the system manual is an overview page that lists the title of each chapter. Following this page, you will find a table of contents.
- At the beginning of each chapter is a table of contents for that chapter. Each chapter has three level headings that are numbered. The fourth level heading is not numbered but appears in **boldface type**.
- Pages, figures, and tables are numbered separately for each chapter. On the back of the table of contents for each chapter you will find a list of the figures and tables that appear in that chapter.

This system manual employs the following specific conventions:

- Specific terms have characteristic abbreviations (e. g. programmer is PG).
- Footnotes are marked with a raised number (e. g. "^{1}") or a raised asterisk ("*"). You will find the corresponding explanations in the lower margin of the page.
  Lists are designated with bullets (•) as in this particular listing or with hyphens (-).
  Procedures are marked with black triangles ( ).
- Cross references are indicated as follows: "(see section 7.3.2)".
  There are no references to specific page numbers.
- Dimensions in drawings are indicated in millimeters and inches.
- Value ranges are indicated as follows: 17 to 21.

You will find definitions for the terms "Warning" and "Note" in the Safety-Related Guidelines for the User at the end of the introduction.

Safety-Related Guidelines for the User

This document provides the information required for the intended use of the particular product. The documentation is written for technically qualified personnel.

Qualified personnel as referred to in the safety guidelines in this document as well as on the product itself are defined as follows.

- System planning and design engineers who are familiar with the safety concepts of automation equipment.
- Operating personnel who have been trained to work with automation equipment and are conversant with the contents of the document in as far as it is connected with the actual operation of the plant.
- Commissioning and service personnel who are trained to repair such automation equipment and who are authorized to energize, de-energize, clear, ground, and tag circuits, equipment, and systems in accordance with established safety practice.

Danger Notices

The notices and guidelines that follow are intended to ensure personal safety, as well as protect the products and connected equipment against damage.

The safety notices and warnings for protection against loss of life (the users or service personnel) or for protection against damage to property are highlighted in this document by the terms and pictograms defined here. The terms used in this document and marked on the equipment itself have the following significance.

Note

contains important information about the product, its operation or a part of the document to which special attention is drawn.

Warning

indicates that death, severe personal injury or substantial property damages can result if proper precautions are not taken.

Proper Usage

Warning

- The equipment/system or the system components may only be used for the applications described in the catalog or the technical description, and only in combination with the equipment, components, and devices of other manufacturers as far as this is recommended or permitted by Siemens.
- The product will function correctly and safely only if it is transported, stored, set up, and installed as intended, and operated and maintained with care.

1 Overview and Installation of COM 95F

Figures

Tables

1 Overview and Installation of COM 95F

COM 95F is a software package specially created for the safety-related S5-95F programmable controller system. COM 95F supports you in the following:

- Entering the system parameters
- Documenting the system parameterization
- Analysing the system messages of the S5-95F

The COM 95F software package has a context sensitive Help system which provides you with further information when desired at almost any point. You can access the Help system by pressing <Ctrl B> or <Strg B>.

The following description is a general guideline to help you initially in working with COM 95F. For any support in working with COM 95F just pres tha HELP key.

1.1 Scope of Supply of COM 95F

The COM 95F software package is supplied on two 3.5" floppy disks. The first disk is required when working with with a STEP 5 basic package up to Version V 6.x; the second disk is required when working with a STEP 5 basic package Version V 7.0 upwards.

Contents of the COM 95F software package

Besides the COM 95F configuration tool, the disks contain further tools which are useful for plant acceptance and output of error messages.

Table 1-1. Contents of COM 95F Software Package

Component	Remark
COM 95F configuration tool	For installation and description, see Chapters 2 to 4
Change comparator and logical comparator for plant acceptance	For installation and description, see Chapter 5
File for output of error messages to printer	The error messages of the S5-95F can be printed out in plain text using the CP 521 communication module. The `CP521DST.S5D` file on the disk contains a data block with the relevant texts. For details, see S5-95F manual, Section 15.5.
File for output of error messages on OP 15 Operator Panel	The error messages of the S5-95F can also be output in plain text on the operator panel. The `FB235D.015` file on the disk contains the relevant texts configured with ProTool. For details, see S5-95F manual, Section 15.5.

The following section describes installation of the software on your programmer/PC. We assume that the floppy disk drive is defined as A: and the hard disk as C:.

Installing COM 95F on your Programmer or PC with STEP 5 Basic Package up to Version V 6.x

If you are operating with a STEP5 basic package Version V 3.x or V 6.x, proceed as described in the following table.

Table 1-2. Installing COM 95F on a Programmer/PC with STEP 5 Basic Package up to Version V 6.x

Step	Action	Response
1	Insert the disk in drive A:	
2	Select drive A: and call up the installation program `INST_F.EXE`	The opening menu of the installation program appears on the monitor
3	Select the COM 95F package in the installation selection and answer the questions on the screen	The parameterization software COM 95F is automatically installed. When the installation is complete, the following message box appears on the screen: "COM 95F successfully installed"

Installing COM 95F on your Programmer or PC with STEP 5 Basic Package Version V 7.0 Upwards

If you are using a STEP5 basic package Version V 7.0 upwards, proceed as described in the following table.

Table 1-3. Installing COM 95F on a Programmer/PC with STEP 5 Basic Package Version V 7.0 Upwards

Step	Action	Response
1	Insert the disk in drive A:	
2	Select drive A: and call up the installation program `INSTALL.EXE`	The opening menu of the installation program appears on the monitor
3	Select the COM 95F package in the installation selection and answer the questions on the screen	The parameterization software COM 95F is automatically installed. When the installation is complete, the following message box appears on the screen: "COM 95F successfully installed"

1.2 Starting COM 95F

After installing COM 95F, you can start the package in the usual manner. COM 95F requires 580 KByte of free DOS memory in order to execute properly.

After entering the obligatory Defaults screen form (see programmer manual), COM 95F branches to the main menu.

Depending on your STEP 5 Basic Package update, COM 95F will present different start screen forms. Below is an overview of the main menu, the functions of COM 95F and the operating hierarchy.

Functions selectable in COM 95F

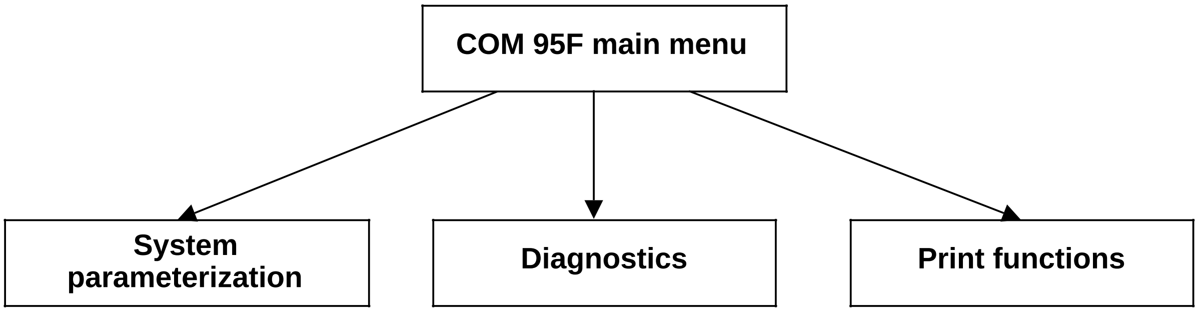
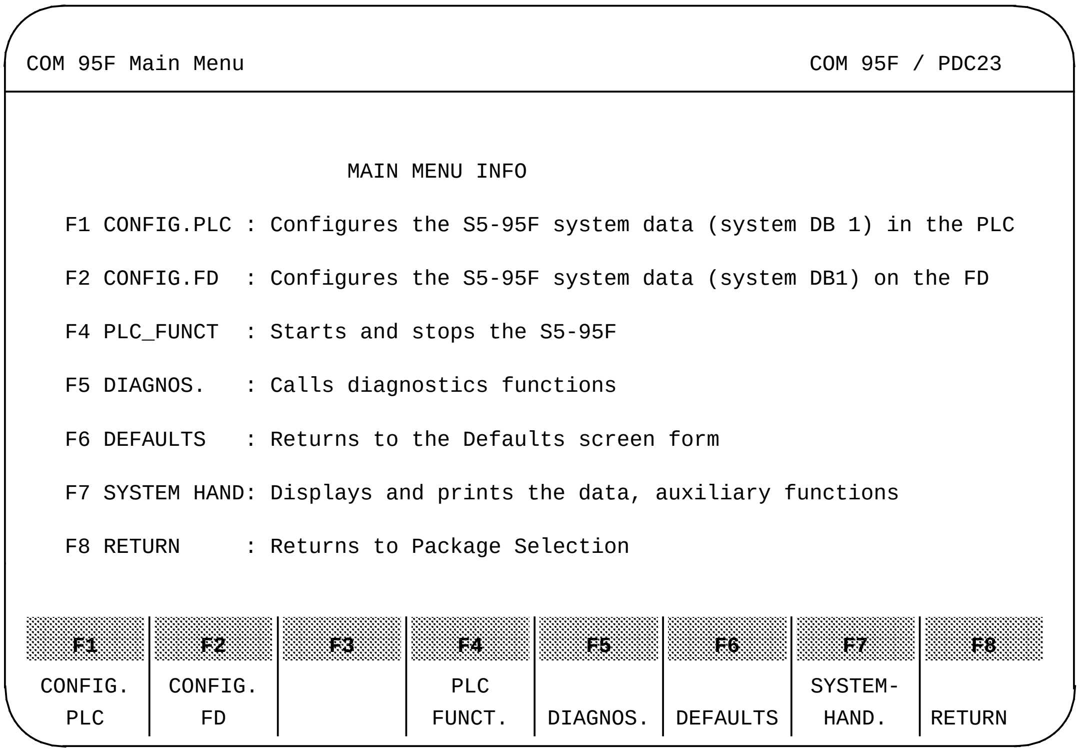


Figure 1-1. COM 95F Functions

1.2.1 Operating Hierarchy of COM 95F under STEP 5 to V 3.x

Figure 1-2 shows a COM 95F start screen form example under STEP 5 to V 3.x.

Start screen form for COM 95F under STEP 5 to V 3.x



```
COM 95F Main Menu                                             COM 95F / PDC23

                         MAIN MENU INFO

   F1 CONFIG.PLC : Configures the S5-95F system data (system DB 1) in the PLC

   F2 CONFIG.FD  : Configures the S5-95F system data (system DB1) on the FD

   F4 PLC_FUNCT  : Starts and stops the S5-95F

   F5 DIAGNOS.   : Calls diagnostics functions

   F6 DEFAULTS   : Returns to the Defaults screen form

   F7 SYSTEM HAND: Displays and prints the data, auxiliary functions

   F8 RETURN     : Returns to Package Selection

    F1        F2        F3        F4        F5        F6        F7        F8
  CONFIG.   CONFIG.             PLC                           SYSTEM-
    PLC       FD               FUNCT.   DIAGNOS.  DEFAULTS    HAND.    RETURN
```

Figure 1-2. Example: Start Screen Form for COM 95F under STEP 5 to V 3.x

Table 1-4 shows the menu points under which the individual functions can be accessed.

Table 1-4. A Selection of Functions Available in STEP 5 to V 3.x

Function	Selecting the Function in the Main Menu
System configuration in the S5-95F (on-line parameterization)	Press <F1> if you want to enter or modify the system parameterization direct in the S5-95F.
System configuration on FD	Press <F2> if you want to enter or modify the system parameterization on the disk.
PLC functions	Press <F4> if you want to start or stop the S5-95F via the programmer
Diagnostics and error messages	Press <F5> if you want to evaluate the error messages and/or the static error images.
Printer functions	Press <F7> if you want to document and print the system parameterization.

1.2.2 Operating Hierarchy of COM 95F under STEP 5, V 6.x

If you load COM 95F under the menu point "Change", the package displays the start screen form. Figure 1-3 shows an example.

Start screen form for COM 95F under STEP 5, V 6.x

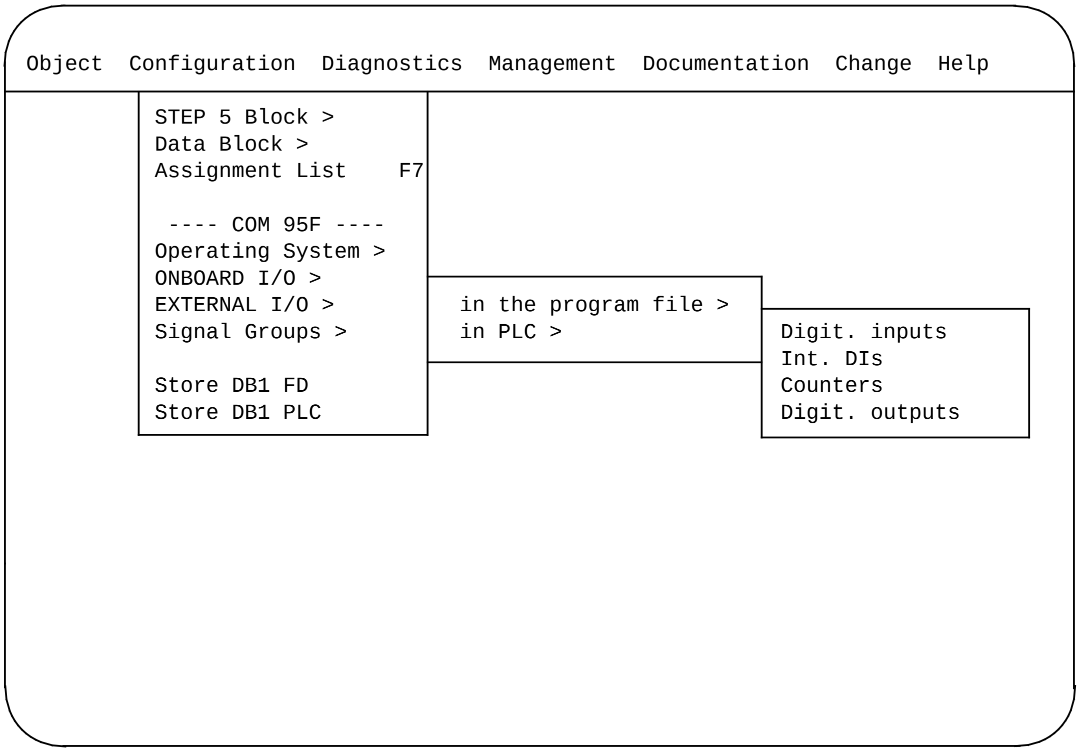


Figure 1-3. Example: Start Screen Form for COM 95F under STEP 5, V 6.x

The following table shows a selection of the functions available in the Drop-Down Menu (DDM).

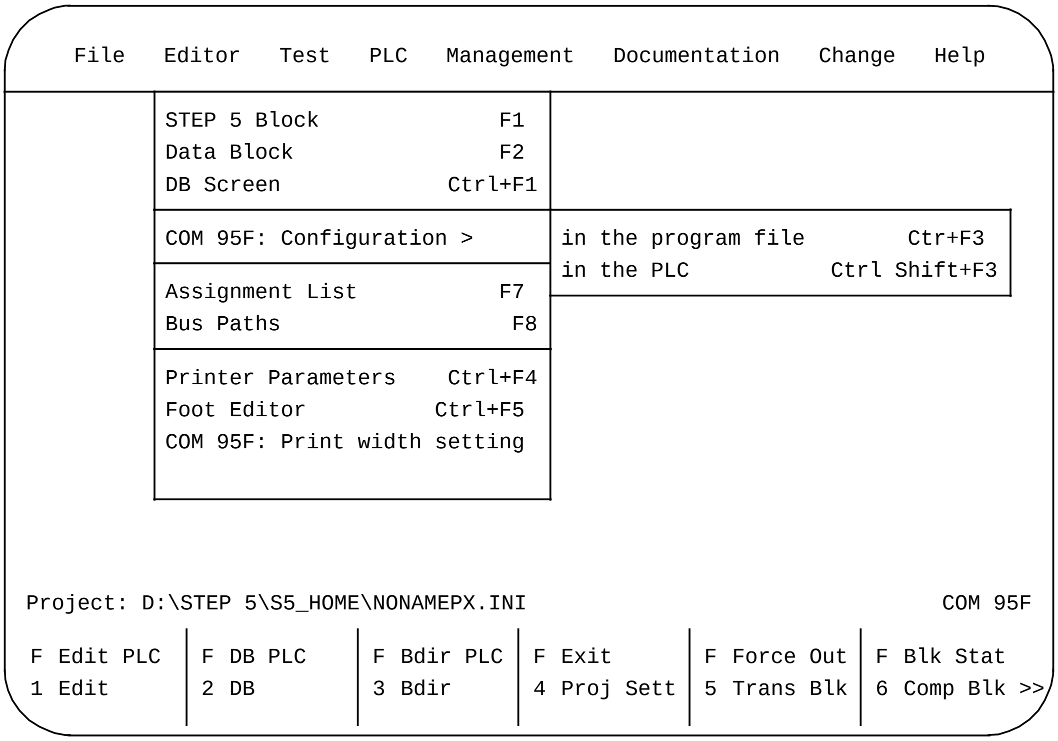
Table 1-5. A Selection of Functions Available in STEP 5, Version V 6.x Upwards

Function	Selecting the Function in the Main Menu
Edit system configuration - in the S5-95F (online) - in the program file (offline)	Select the desired parameter range in the **Configuration** DDM if you want to enter or modify system parameters.
Store system configuration - in the S5-95F (online) - in the program file (offline)	Select "Store DB 1" in the **Configuration** DDM if you want to store the system parameterization.
Diagnostics and error messages	Select the relevant line in the **Diagnostics** DDM if you want to evaluate error messages and/or static error images.
Print functions	Select the relevant line in the **Documentation** DDM if you want to document and print out the system parameterization.

1.2.3 Operating Hierarchy of COM 95F under STEP 5 from V 7.0

If you load COM 95F under the menu item "Change", the package displays the start screen form. Figure 1-4 shows an example.

Start screen form for COM 95F under STEP 5 V 7.0



Project: D:\STEP 5\S5_HOME\NONAMEPX.INI

Figure 1.4. Example: Start Screen Form for COM 95F under STEP 5 from V 7.0

The following table shows a selection of the functions available in the Drop-Down Menu (DDM).

Table 1-6. A Selection of Functions Available in STEP 5 from V 7.0

Function	Selecting the Function in the Main Menu
Edit system configuration - in the S5-95F (online) - in the program file (offline)	Select the desired range in the **Editor** DDM under menu item COM 95F: Configuration > if you want to enter or modify system parameters.
Store system configuration - in the S5-95F (online) - in the program file (offline)	Select "Store DB 1" in the **File** DDM under **Configuration** if you want to store the system parameterization.
Diagnostics and error messages	Select the relevant line in the **PLC** DDM if you want to evaluate error messages and/or static error images.
Print functions	Select the relevant line in the **Documentation** DDM if you want to document and print out the system parameterization.

2 Entering the System Parameters for DB1

Figures

Tables

2 Entering the System Parameters for DB1

You must inform the operating system of a range of parameters depending on the application and configuration of the S5-95F. These parameters are stored in DB 1 of the S5-95F as ASCII text.

The COM 95F parameterization software has been created to save you the trouble of editing DB 1. You simply fill in some user-friendly screen forms in interactive dialog with COM 95F and the software then generates DB1 with the required parameters.

COM 95F handles the following security tasks:

- Check that the entered system parameters are valid and permissible
- Check that DB 1 is compiled correctly

System parameterization procedure

The next pages show you the most effective procedure for entering parameters.

We suggest the following order but you may of course deviate from this:

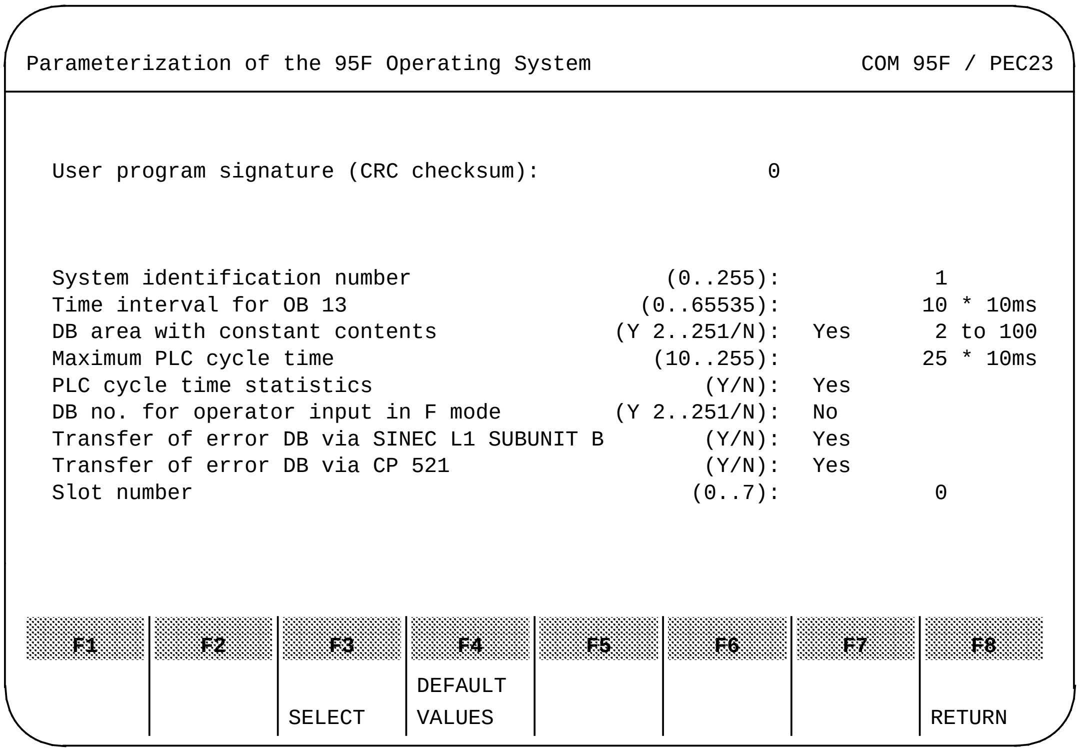
Parameterization of the general operating system data
Parameterization of the integral clock
Parameterization of SINEC L1 communications
Parameterization of the I/O
Parameterization of the system response to I/O failure

The COM 95F software package has a context-sensitive Help system which provides you with further information as desired at almost any point. For this reason, the following description should serve only as a guide to make the first steps with COM 95F easier.

2.1 Parameterizing the General Operating System Data

The S5-95F requires some general system-dependent parameters. You must enter these parameters in the following screen form.

Example: Parameterizing the general operating system data



```
Parameterization of the 95F Operating System                   COM 95F / PEC23

  User program signature (CRC checksum):                  0

  System identification number                  (0..255):             1
  Time interval for OB 13                     (0..65535):            10 * 10ms
  DB area with constant contents            (Y 2..251/N):  Yes        2 to 100
  Maximum PLC cycle time                       (10..255):            25 * 10ms
  PLC cycle time statistics                        (Y/N):  Yes
  DB no. for operator input in F mode       (Y 2..251/N):  No
  Transfer of error DB via SINEC L1 SUBUNIT B      (Y/N):  Yes
  Transfer of error DB via CP 521                  (Y/N):  Yes
  Slot number                                    (0..7):              0
```

Figure 2-1. Example: Parameterization of the 95F Operating System

Parameterization notes

Table 2-1. Parameterizing the 95F Operating System

Parameters	Remarks
`User program signature (CRC checksum):`	The S5-95F enters the user program signature autonomously as soon as the S5-95F is switched from STOP to RUN.
`System identification number`	If you are using several S5-95Fs, enter a different number here for each S5-95F (see S5-95F manual).
`Time interval for OB 13`	Enter here the OB 13 interval if you are using time-controlled program processing or if you are using OB 13-oriented discrepancy times.
`DB area with constant contents`	Enter here the block numbers of the DBs which do not change even in operation. DBs with constant contents should be used in a contiguous order and are included in the CRC checksum.
`Maximum PLC cycle time`	Enter here the maximum permissible PLC cycle time for your process. Important Entry of the maximum PLC cycle time determines the max. S5-95F response time (see S5-95F manual).
`PLC cycle time statistics`	Enter "YES" here if you want to use the S5-95F statistics function (see S5-95F manual).
`DB no. for operator input in F mode`	Enter here the number of the parameter control DB. In safety-related operation, you can modify only the parameter control DB via the programmer (see S5-95F manual). This allows you to pass modifiable data to the user program. However, these data must be checked for plausibility.
`Transfer of error DB via SINEC L1 subunit B`	Enter "YES" here if you want to send the coded system messages to the SINEC L1 master (see S5-95F manual).
`Transfer of error DB via CP 521`	Enter "YES" here if you want to print the S5-95F messages (see S5-95F manual).
`Slot number`	Enter here the slot number of the CP 521 to which the system messages are to be output.

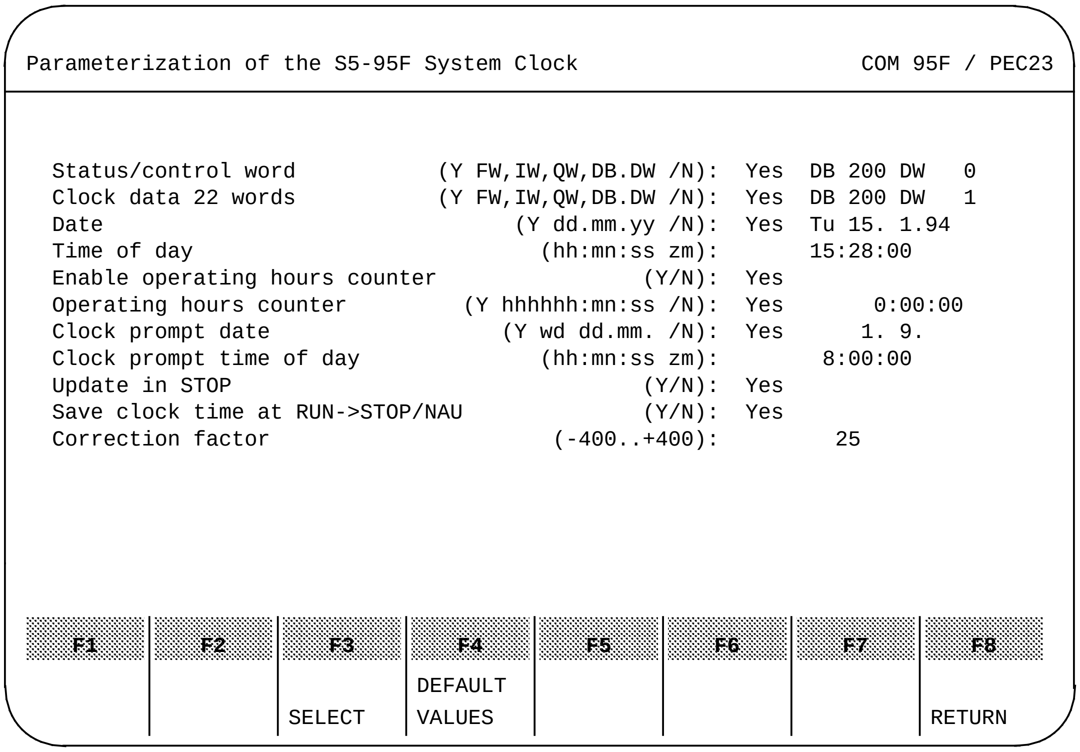
2.2 Parameterizing the Integral Clock

If you want to use the integral clock on the S5-95F, you must parameterize the clock with COM 95F. Please note that you must not initiate safety-related actions from the clock data.

The parameters entered are read by the S5-95F at the first startup. In the case of a battery-backed clock, the clock data is retained and no longer changed.

However, you can change the clock data via the programmer. See Chapter 10 of the S5-95F manual.

Example: Parameterizing the integral clock



```
Parameterization of the S5-95F System Clock                    COM 95F / PEC23

  Status/control word           (Y FW,IW,QW,DB.DW /N):  Yes  DB 200 DW   0
  Clock data 22 words           (Y FW,IW,QW,DB.DW /N):  Yes  DB 200 DW   1
  Date                                (Y dd.mm.yy /N):  Yes  Tu 15. 1.94
  Time of day                           (hh:mn:ss zm):       15:28:00
  Enable operating hours counter                (Y/N):  Yes
  Operating hours counter         (Y hhhhhh:mn:ss /N):  Yes       0:00:00
  Clock prompt date                  (Y wd dd.mm. /N):  Yes      1. 9.
  Clock prompt time of day              (hh:mn:ss zm):        8:00:00
  Update in STOP                                (Y/N):  Yes
  Save clock time at RUN->STOP/NAU              (Y/N):  Yes
  Correction factor                      (-400..+400):         25
```

Figure 2-2. Example: Parameterization of the S5-95F System Clock

Parameterization notes

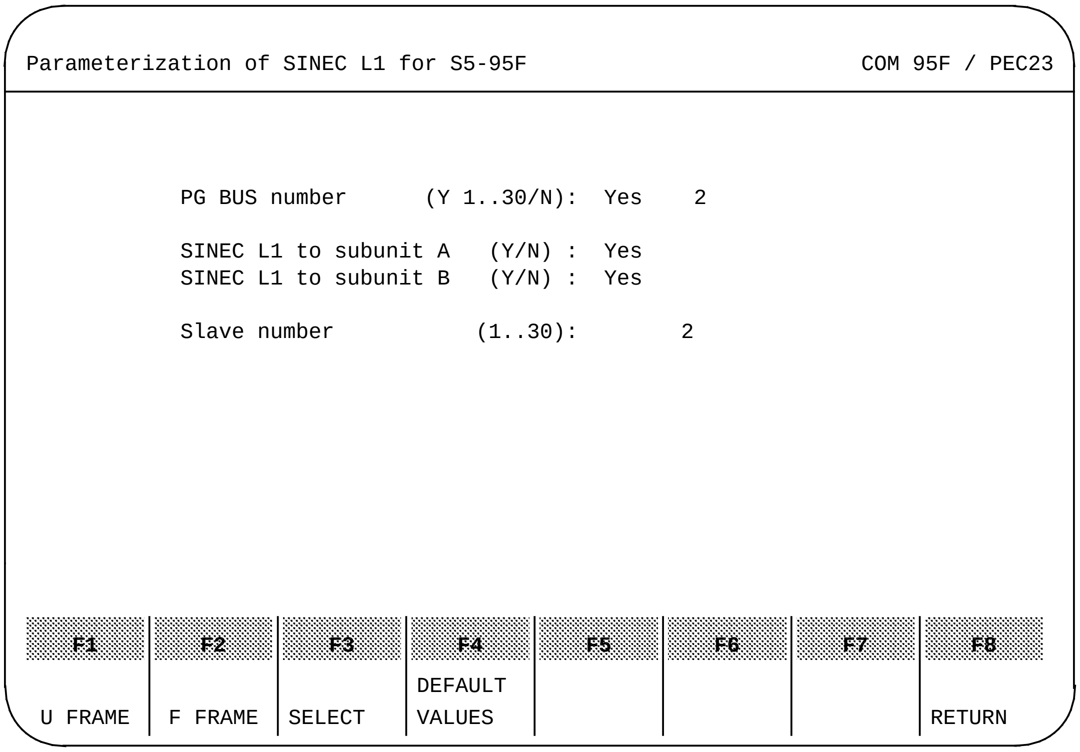
Table 2-2. Parameterizing the System Clock of the S5-95F

Parameters	Remarks
`Status/control word`	Enter the clock status/control word here. So that you can also set the clock in safety operation, we recommend that you use a word in the parameter control DB.
`Clock data 22 words`	Enter the first word of the clock data area here. The clock data area occupies 22 continuous words. We recommend that you store the clock data area in the parameter control DB.
`Date`	Enter the date here that the S5-95F is to assume at the first startup.
`Time of day`	Enter the time of day here that the S5-95F is to assume at the first startup.
`Enable operating hours counter`	Enter "Yes" here if the S5-95F operating hours counter is also to be captured.
`Operating hours counter`	Enter the starting value for the operating hours counter here.
`Clock prompt date`	Enter the date for the prompt monitor here.
`Clock prompt time of day`	Enter the time of day for the prompt monitor here.
`Update in STOP`	Enter "Yes" here if the internal clock is to continue operating even in the case of S5-95F STOP.
`Save time of day at RUN->STOP/NAU`	Enter "Yes" here if the time of day is to be stored at the transition from RUN to STOP or in the event of power failure in the clock data area.
`Correction factor`	Enter the correction factor here if you want to alter the accuracy.

2.3 Parameterizing SINEC L1 Communications

You must differentiate between safety-related data exchange and non-safety-related data exchange when parameterizing SINEC L1. Parameterization of SINEC L1 is described below using some examples.

Example: Assigning the general SINEC L1 parameters



```
Parameterization of SINEC L1 for S5-95F                          COM 95F / PEC23


            PG BUS number      (Y 1..30/N):  Yes    2

            SINEC L1 to subunit A   (Y/N) :  Yes
            SINEC L1 to subunit B   (Y/N) :  Yes

            Slave number           (1..30):        2
```

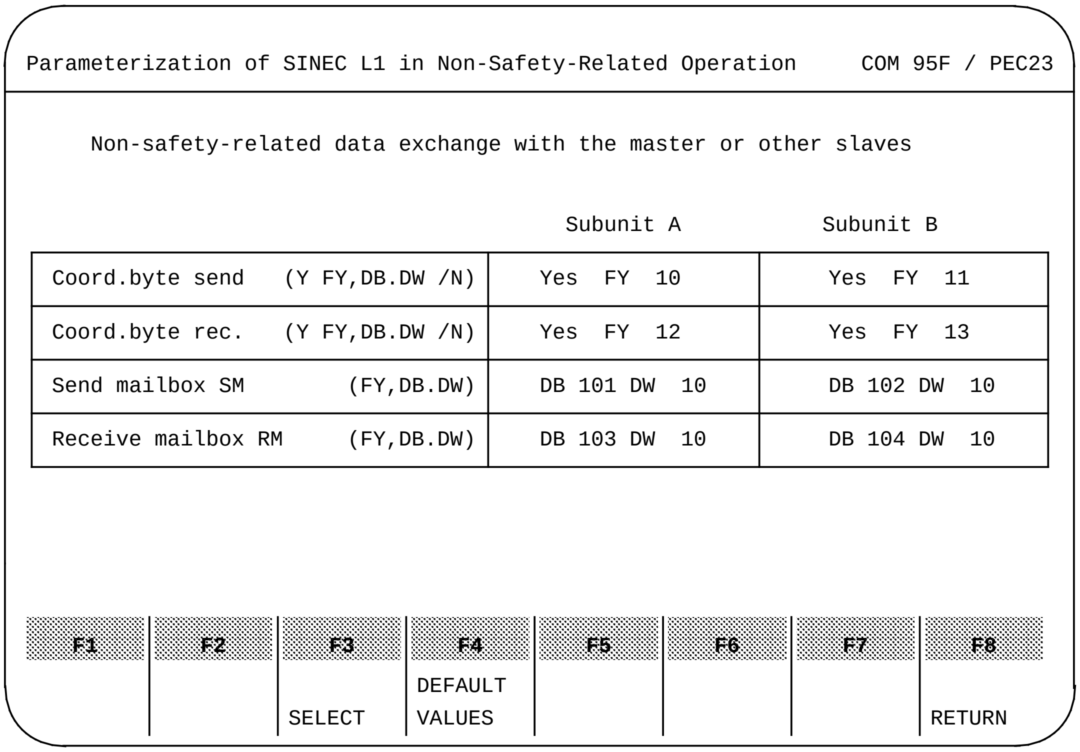
Figure 2-3. Example: Parameterization for Activating the SINEC L1 LAN

Parameterization notes

Table 2-3. Parameterization for Activating the SINEC L1 LAN

Parameters	Remarks
`PG BUS number`	Enter the PG Bus number here if you want to use the PG bus (see SINEC L1 manual).
`SINEC L1 to subunit A`	Enter "Yes" here if you are connecting subunit A to SINEC L1.
`SINEC L1 to subunit B`	Enter "Yes" here if you are connecting subunit B to SINEC L1.
`Slave number`	Enter the node number of the S5-95F here.

Example: Assigning SINEC L1 parameters for non-safety-related data exchange



Parameterization of SINEC L1 in Non-Safety-Related Operation COM 95F / PEC23

Non-safety-related data exchange with the master or other slaves

	Subunit A	Subunit B
Coord.byte send (Y FY,DB.DW /N)	Yes FY 10	Yes FY 11
Coord.byte rec. (Y FY,DB.DW /N)	Yes FY 12	Yes FY 13
Send mailbox SM (FY,DB.DW)	DB 101 DW 10	DB 102 DW 10
Receive mailbox RM (FY,DB.DW)	DB 103 DW 10	DB 104 DW 10

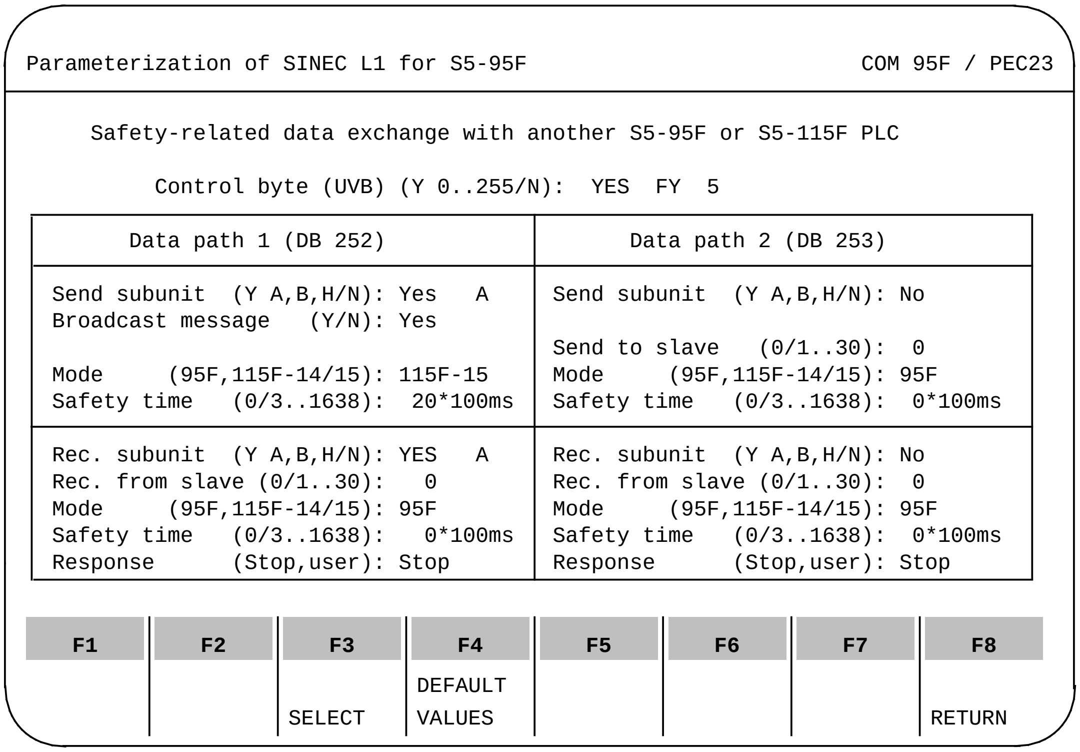
Figure 2-4. Example: Parameterization of SINEC L1 in Non-Safety-Related Operation

Parameterization notes

Table 2-4. Parameterization of SINEC L1 in Non-Safety-Related Operation

Parameters	Remarks
Cord. byte send	Enter here the byte in which the coordination information for sending is to be stored.
Coord. byte rec.	Enter here the byte in which the coordination information for receiving is to be stored.
Send mailbox SM	Enter the desired location of the first word of the Send mailbox here.
Receive mailbox RM	Enter the first word of the Receive mailbox here.

Example: Assigning SINEC L1 parameters for safety-related data exchange



```
Parameterization of SINEC L1 for S5-95F                         COM 95F / PEC23

     Safety-related data exchange with another S5-95F or S5-115F PLC

          Control byte (UVB) (Y 0..255/N):  YES  FY  5

        Data path 1 (DB 252)                    Data path 2 (DB 253)

 Send subunit  (Y A,B,H/N): Yes   A     Send subunit  (Y A,B,H/N): No
 Broadcast message   (Y/N): Yes
                                        Send to slave   (0/1..30):  0
 Mode     (95F,115F-14/15): 115F-15     Mode     (95F,115F-14/15): 95F
 Safety time   (0/3..1638):  20*100ms   Safety time   (0/3..1638):  0*100ms

 Rec. subunit  (Y A,B,H/N): YES   A     Rec. subunit  (Y A,B,H/N): No
 Rec. from slave (0/1..30):   0         Rec. from slave (0/1..30):  0
 Mode     (95F,115F-14/15): 95F         Mode     (95F,115F-14/15): 95F
 Safety time   (0/3..1638):   0*100ms   Safety time   (0/3..1638):  0*100ms
 Response      (Stop,user): Stop        Response      (Stop,user): Stop

   F1       F2       F3       F4       F5       F6       F7       F8
                             DEFAULT
                    SELECT   VALUES                               RETURN
```

Figure 2-5. Example: Parameterization of SINEC L1 for S5-95F

Parameterization notes

Table 2-5. Parameterization of SINEC L1 for S5-95F

Parameters	Remarks
Control byte (UVB)	Enter here the byte in which the coordination information for safety-related data exchange is to be stored.
Send subunit	Enter the subunit from which you want to send message frames here.
Broadcast message	Enter here whether you want to send safety-related broadcast messages via data path 1. (If you select Broadcast you cannot send data via data path 2.)
Send to slave	Enter here the node number of the slave to which you want to send data.
TYPE	Specify whether the communications partner is an S5-95F or an S5-115F.
Mode	Enter here the message mode to be used by the S5-95F (see S5-95F manual, Section 13).

Table 2-5. Parameterization of SINEC L1 for S5-95F (Continued)

Parameters	Remarks
`Safety time`	Enter the SINEC L1 safety time here. At least one valid message frame must be sent/received within this safety time.
`Response`	Enter here whether the S5-95F is to respond to a SINEC L1 fault with STOP or with passivation of the data path.
`Rec. subunit`	Enter here the subunit from which you want to receive message frames.
`Rec. from slave`	Enter here the node number of the slave from which you want to receive data.

2.4 Parameterizing the I/O

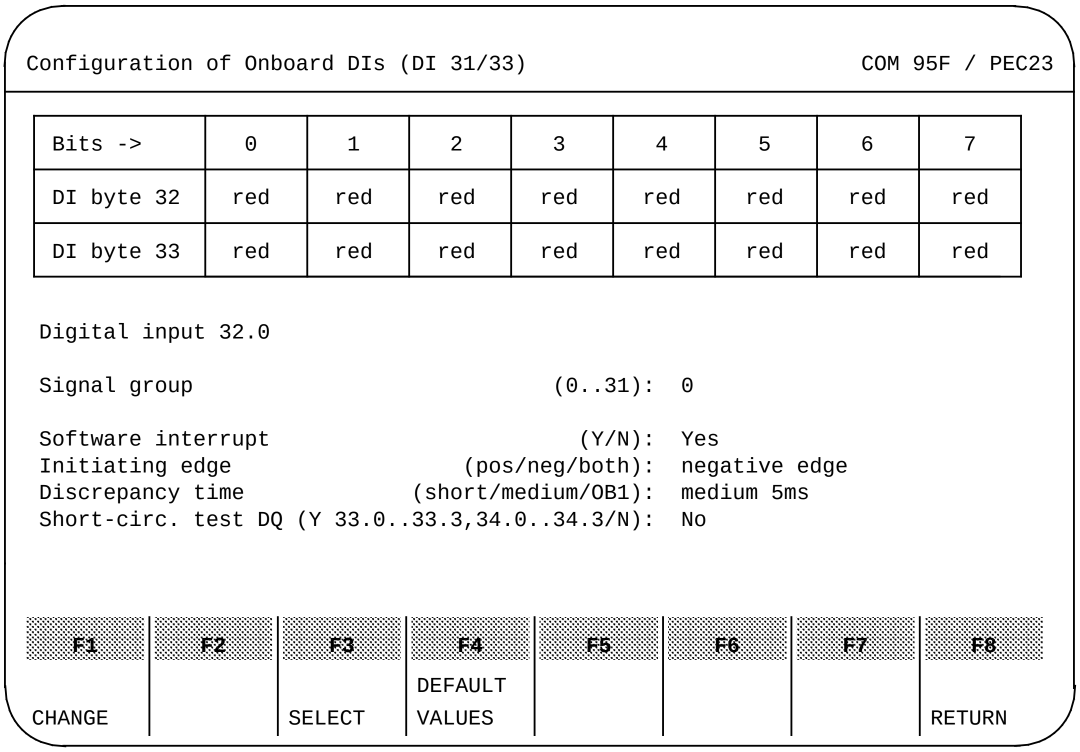
The largest part of system parameterization is taken up with assigning I/O parameters. Below, we explain parameterization of the following, using some examples:

- Onboard DIs (DI byte 32 ... 33)
- Onboard interrupt DIs (DI byte 59)
- Onboard DQs (DQ byte 32)
- Onboard counters
- External I/O
- Signal groups (system response to I/O failure)

2.4.1 Parameterizing the Onboard DIs (DI Byte 32, 33)

The onboard DIs (DI byte 32, 33) can be parameterized either as software interrupts (OB3 interrupt DI) or as failsafe standard DIs.

Example: Parameterizing the onboard DIs (DI 32, 33)



Configuration of Onboard DIs (DI 31/33) COM 95F / PEC23

Bits ->	0	1	2	3	4	5	6	7
DI byte 32	red	red	red	red	red	red	red	red
DI byte 33	red	red	red	red	red	red	red	red

```
Digital input 32.0

Signal group                           (0..31):  0

Software interrupt                       (Y/N):  Yes
Initiating edge                  (pos/neg/both):  negative edge
Discrepancy time             (short/medium/OB1):  medium 5ms
Short-circ. test DQ (Y 33.0..33.3,34.0..34.3/N):  No
```

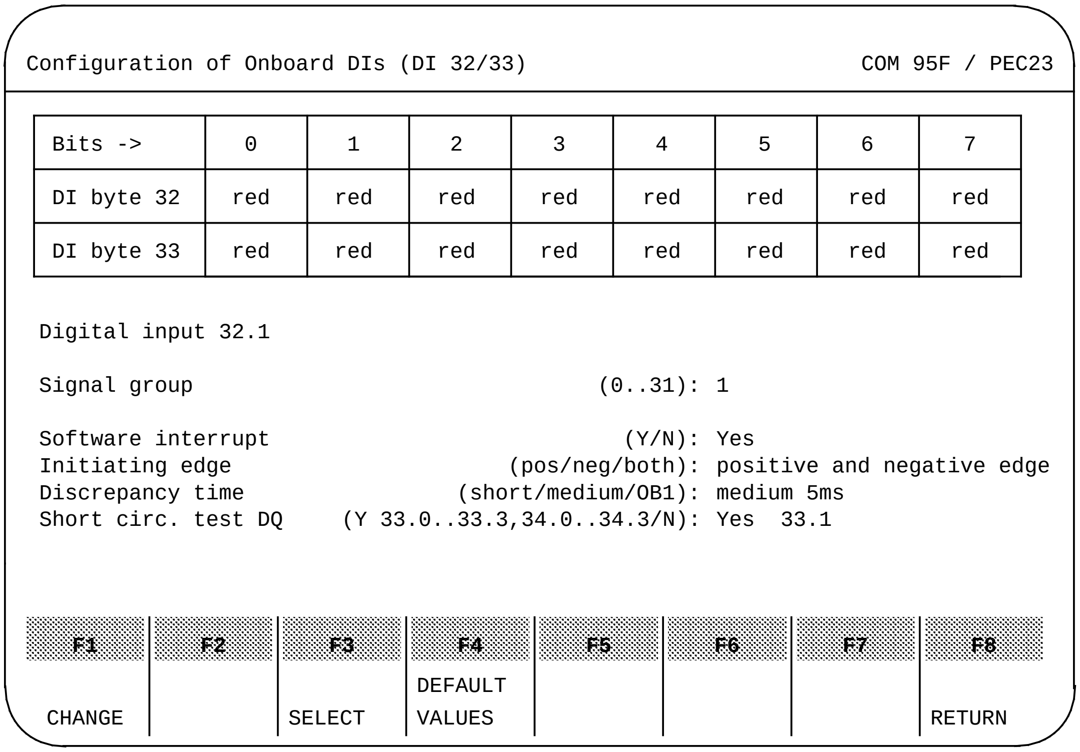
Figure 2-6. Example: Configuring the Onboard DIs (DI 32/33)

Parameterization notes

Table 2-6. Configuring the Onboard DIs (DI 32/33)

Parameters	Remarks
`Signal group`	Enter here the number of the signal group to which you want to assign the I/O. The assignment is significant for the system response to I/O failure (see Section 2.5).
`Software interrupt`	Enter "Yes" here if the selected bit is used as software interrupt DI. Enter "No" here if the selected bit is used as safety-related standard DI.
`Initiating edge`	Enter here the type of edge that is to initiate an interrupt (see S5-95F manual, Section 12.3)
`Discrepancy time`	Enter here the period during which the system is to tolerate different input values in subunit A and subunit B.
`Short-circ. test DQ`	If you want to use the line monitor function, enter here the DQ bit via which the input bit is supplied and clocked during the short-circuit test.

Example: Parameterizing a software interrupt DI (DI byte 32, 33)



```
Configuration of Onboard DIs (DI 32/33)                          COM 95F / PEC23

  Bits ->       0     1     2     3     4     5     6     7
  DI byte 32   red   red   red   red   red   red   red   red
  DI byte 33   red   red   red   red   red   red   red   red

 Digital input 32.1

 Signal group                              (0..31): 1

 Software interrupt                          (Y/N): Yes
 Initiating edge                    (pos/neg/both): positive and negative edge
 Discrepancy time               (short/medium/OB1): medium 5ms
 Short circ. test DQ    (Y 33.0..33.3,34.0..34.3/N): Yes  33.1

    F1        F2        F3        F4        F5        F6        F7        F8
                                DEFAULT
  CHANGE              SELECT    VALUES                                  RETURN
```

Figure 2-7. Example: Configuring the Onboard DIs (DI 32/33) as Software Interrupt DIs

Parameterization notes

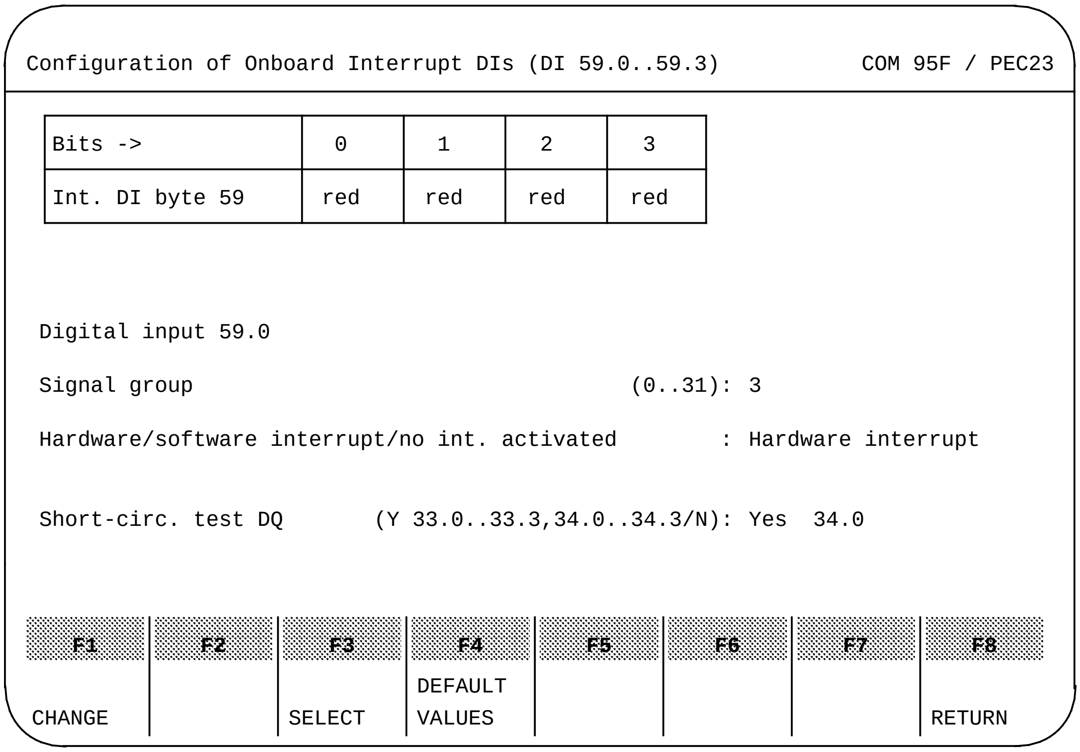
Table 2-7. Configuring the Onboard DIs (DI 32/33) as Software Interrupt DIs

Parameters	Remarks
`Signal group`	Enter here the number of the signal group to which you want to assign the I/Os. The assignment is significant for the system response to I/O failure (see Section 2.5).
`Software interrupt`	Enter "Yes" here if the selected bit is used as software interrupt DI. Enter "No" here if the selected bit is used as safety-related standard DI.
`Initiating edge`	Enter here the type of edge that is to initiate an interrupt (see S5-95F manual, Section 12.3)
`Discrepancy time`	Enter here the period for which the system is to tolerate different input values in subunit A and subunit B.
`Short-circ. test DQ`	If you want to use the line monitor function, enter here the DQ bit via which the input bit is supplied and clocked during the short-circuit test.

2.4.2 Parameterizing the Onboard Interrupt DIs (DI Byte 59)

The onboard interrupt DIs (DI byte 59) can be parameterized as hardware interrupt DIs (OB2 interrupt DIs), OB3 interrupt DIs (OB3 interrupt DIs) or as failsafe standard DIs.

Example: Parameterization of a hardware interrupt DI



```
Configuration of Onboard Interrupt DIs (DI 59.0..59.3)          COM 95F / PEC23
```

Bits ->	0	1	2	3
Int. DI byte 59	red	red	red	red

```
Digital input 59.0

Signal group                                       (0..31): 3

Hardware/software interrupt/no int. activated             : Hardware interrupt


Short-circ. test DQ       (Y 33.0..33.3,34.0..34.3/N): Yes  34.0
```

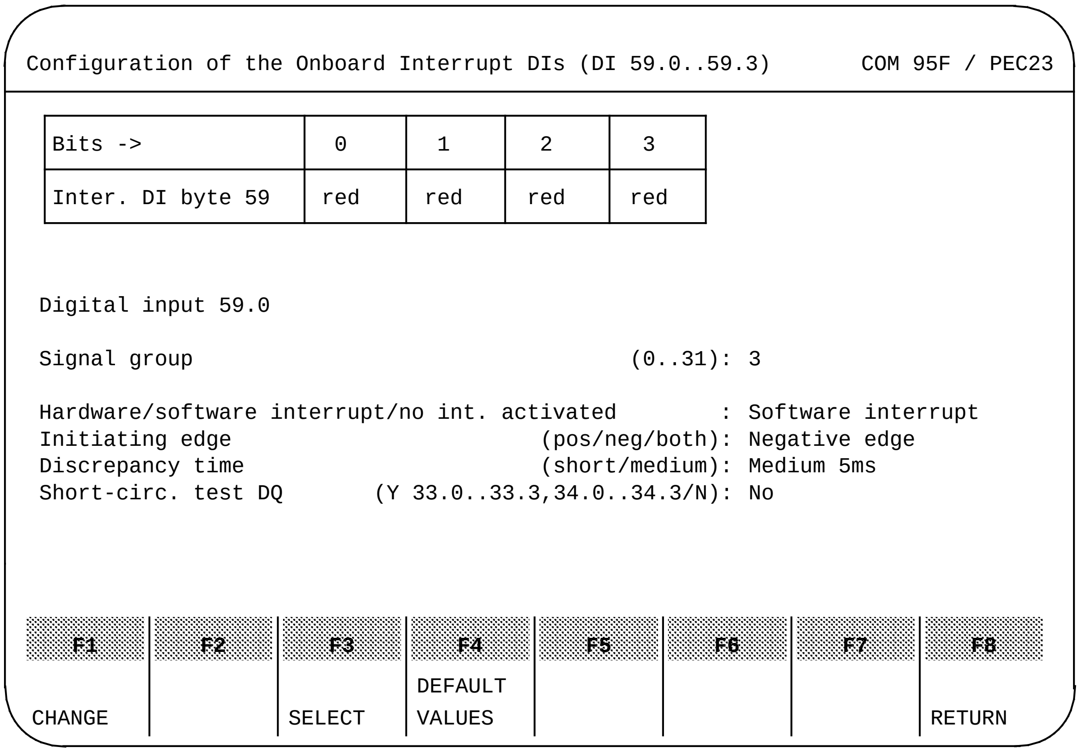
Figure 2-8. Example: Configuration of the Onboard Interrupt DIs (DI 59.0..59.3)

Parameterization notes

Table 2-8. Configuration of the Onboard Interrupt DIs (DI 59.0..59.3)

Parameters	Remarks
Signal group	Enter here the number of the signal group to which you want to assign I/Os. The assignment is significant for the system response to I/O failure (see Section 2.5).
Hardware/software interrupt/no int. activated	Press <F3> to select the task to be performed by the selected DI.
Short-circ. test DQ	If you want to use the line monitor function, enter here the DQ bit via which the input bit is supplied and clocked during the short-circuit test.

Example: Parameterizing a software interrupt DI



Configuration of the Onboard Interrupt DIs (DI 59.0..59.3)   COM 95F / PEC23

Bits ->	0	1	2	3
Inter. DI byte 59	red	red	red	red

```
Digital input 59.0

Signal group                                  (0..31): 3

Hardware/software interrupt/no int. activated        : Software interrupt
Initiating edge                         (pos/neg/both): Negative edge
Discrepancy time                        (short/medium): Medium 5ms
Short-circ. test DQ       (Y 33.0..33.3,34.0..34.3/N): No
```

F1	F2	F3	F4	F5	F6	F7	F8
CHANGE		SELECT	DEFAULT VALUES				RETURN

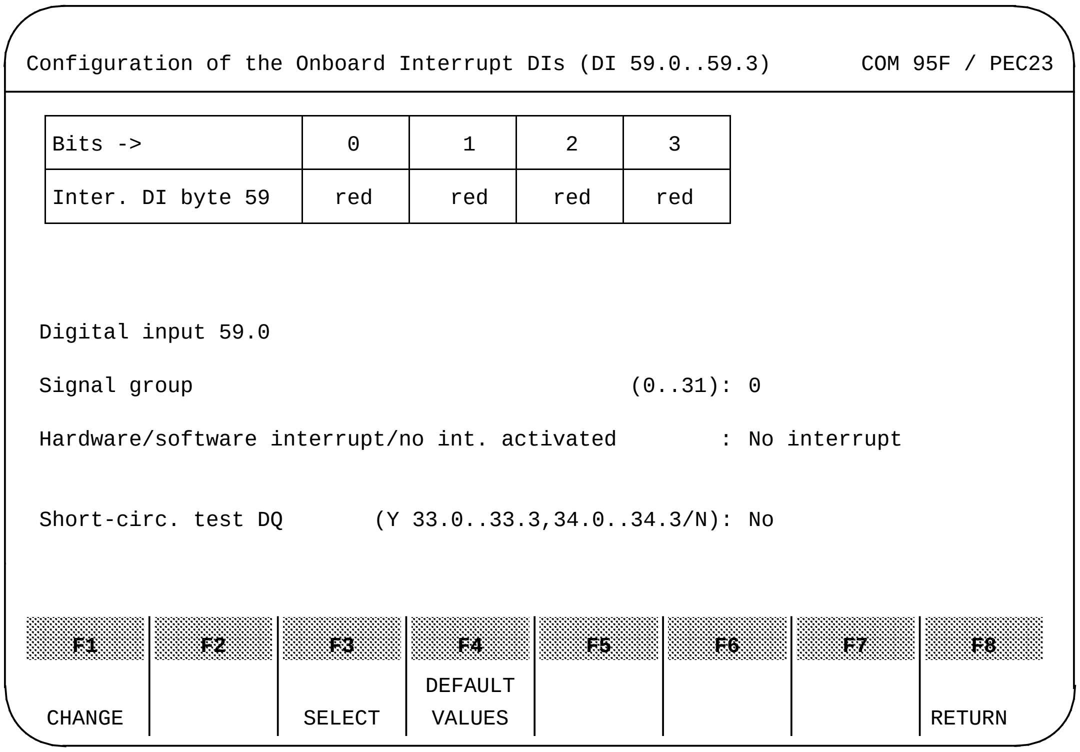
Figure 2-9. Example: Configuration of the Onboard Interrupt DIs (DI 59.0..59.3) as Software Interrupt DIs

Parameterization notes

Table 2-9. Configuration of the Onboard Interrupt DIs (DI 59.0..59.3) as Software Interrupt DIs

Parameters	Remarks
Signal group	Enter here the number of the signal group to which you want to assign I/Os. The assignment is significant for the system response to I/O failure (see Section 2.5).
Hardware/software interrupt/no int. activated	Press <F3> to select the task to be performed by the selected DI.
Initiating edge	Enter here the type of edge that is to initiate an interrupt (see S5-95F manual, Section 12.3)
Discrepancy time	Enter here the period for which the system is to tolerate different input values in subunit A and subunit B.
Short-circ. test DQ	If you want to use the line monitor function, enter here the DQ bit via which the input bit is supplied and clocked during the short-circuit test.

Example: Parameterizing a standard DI



Configuration of the Onboard Interrupt DIs (DI 59.0..59.3)   COM 95F / PEC23

Bits ->	0	1	2	3
Inter. DI byte 59	red	red	red	red

```
Digital input 59.0

Signal group                                   (0..31): 0

Hardware/software interrupt/no int. activated         : No interrupt


Short-circ. test DQ       (Y 33.0..33.3,34.0..34.3/N): No
```

Figure 2-10. Example: Configuration of the Onboard Interrupt DIs (DI 59.0..59.3) as Standard DIs

Parameterization notes

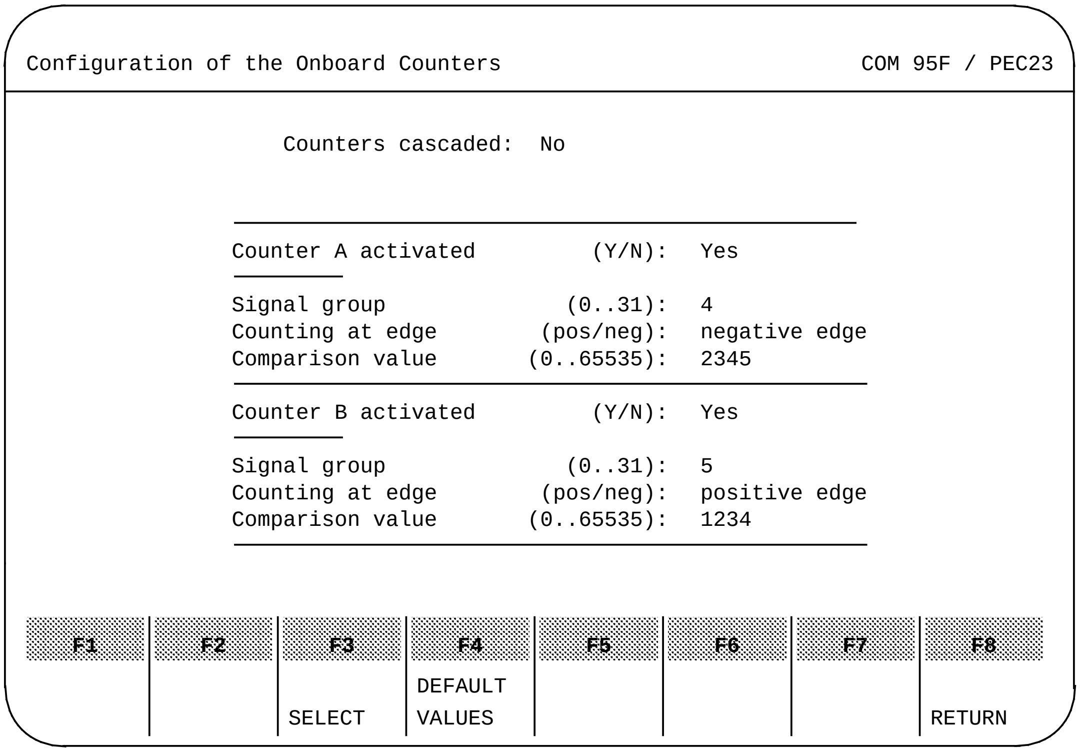
Table 2-10. Configuration of the Onboard Interrupt DIs (DI 59.0..59.3) as Standard DIs

Parameters	Remarks
`Signal group`	Enter here the number of the signal group to which you want to assign I/Os. The assignment is significant for the system response to I/O failure (see Section 2.5).
`Hardware/software interrupt/no int. activated`	Press <F3> to select the task to be performed by the selected DI.
`Short-circ. test DQ`	If you want to use the line monitor function, enter here the DQ bit via which the input bit is supplied and clocked during the short-circuit test.

2.4.3 Parameterizing the Onboard Counters

The S5-95F has two onboard counters. You can use these counters individually or cascaded.

Example: Parameterizing the Onboard Counters



```
Configuration of the Onboard Counters                          COM 95F / PEC23

                     Counters cascaded:  No

                 Counter A activated         (Y/N):   Yes

                 Signal group              (0..31):   4
                 Counting at edge        (pos/neg):   negative edge
                 Comparison value      (0..65535):   2345

                 Counter B activated         (Y/N):   Yes

                 Signal group              (0..31):   5
                 Counting at edge        (pos/neg):   positive edge
                 Comparison value      (0..65535):   1234
```

Figure 2-11. Example: Configuration of the Onboard Counters

Parameterization notes

Table 2-11. Configuration of the Onboard Counters

Parameters	Remarks
`Counters cascaded`	Enter "Yes" here if you want to cascade both onboard counters to form one counter with a large counting range.
`Counter A activated`	Enter "Yes" here if you want to use counter A.
`Counter B activated`	Enter "Yes" here if you want to use counter B.
`Signal group`	Enter here the number of the signal group to which you want to assign I/Os. The assignment is significant for the system response to I/O failure (see Section 2.5).

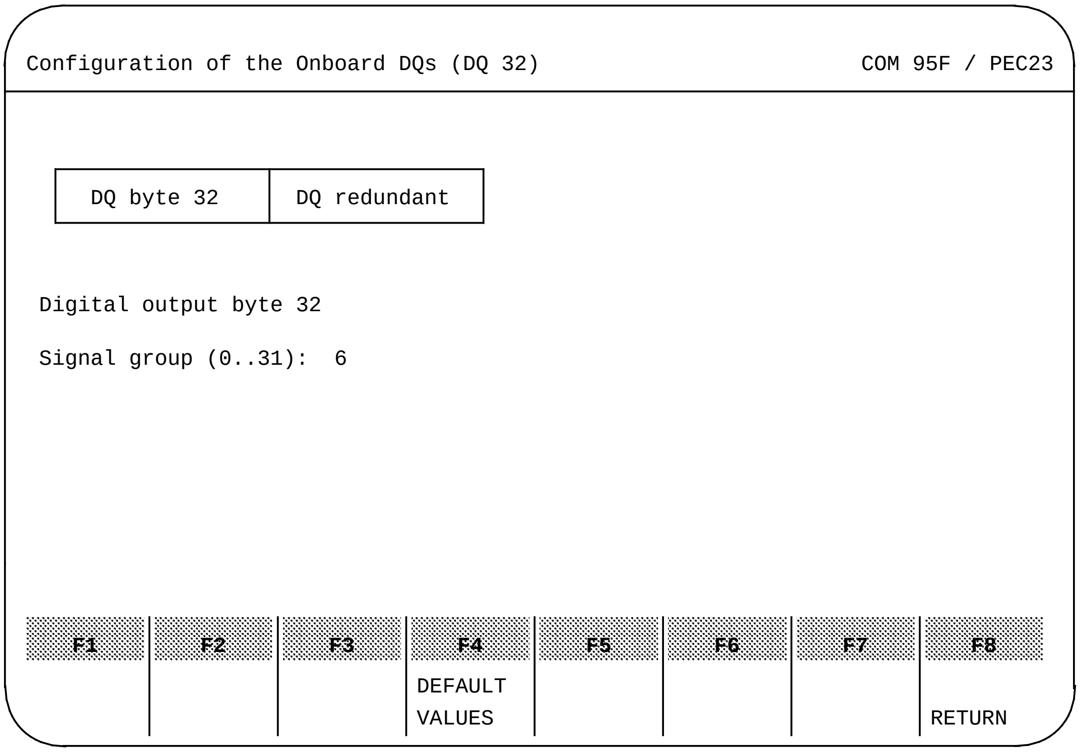
Table 2-11. Configuration of the Onboard Counters (Continued)

Parameters	Remarks
`Counting at edge`	Enter here the type of edge that is to trigger a count pulse.
`Comparison value`	Enter here the comparison value at which the counter is to invoke execution of OB3.

2.4.4 Parameterizing the Onboard DQs

Parameterization of the safety-related onboard DQs is restricted to assignment of the signal group no.

Example: Parameterization of the onboard DQs



```
Configuration of the Onboard DQs (DQ 32)                    COM 95F / PEC23

   DQ byte 32   |  DQ redundant

 Digital output byte 32

 Signal group (0..31):  6
```

Figure 2-12. Example: Configuration of the Onboard DQs (DQ 32)

Parameterization notes

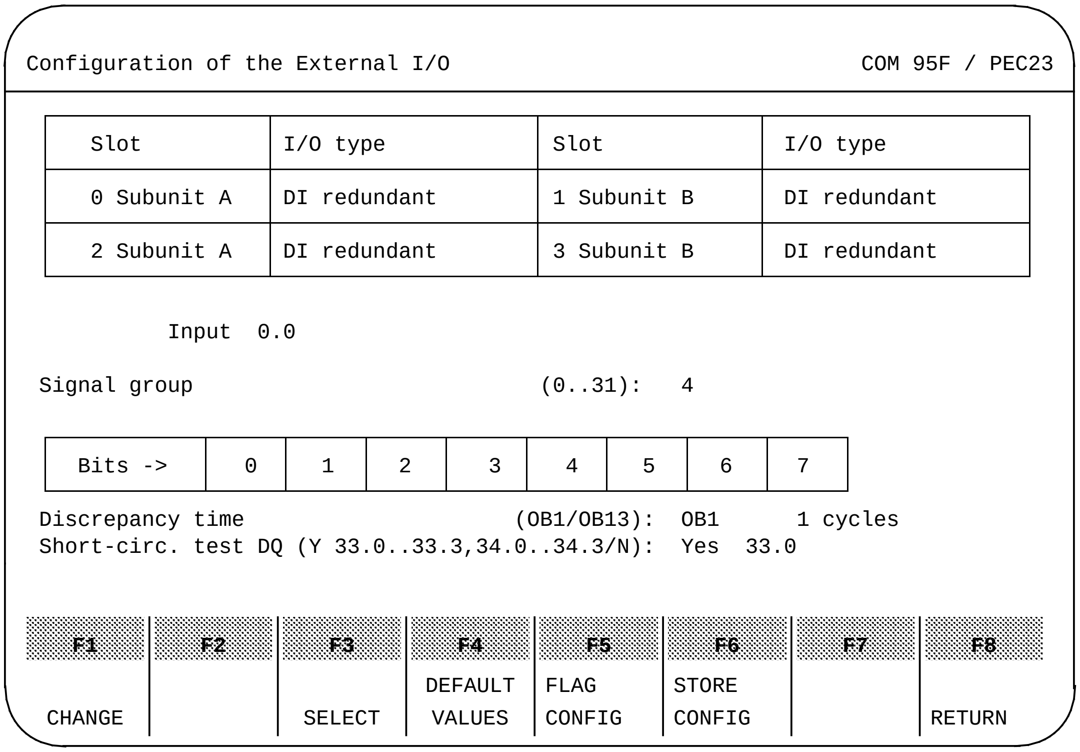
Table 2-12. Configuration of the Onboard DQs (DQ 32)

Parameters	Remarks
`Signal group`	Enter here the number of the signal group to which you want to assign I/Os. The assignment is significant for the system response to I/O failure (see Section 2.5).

2.4.5 Parameterizing the External I/O

All external I/Os are to be parameterized dependent on their slots. Allocate the relevant I/O type used to each slot.

Example: Parameterizing a safety-related DI module



Configuration of the External I/O COM 95F / PEC23

Slot	I/O type	Slot	I/O type
0 Subunit A	DI redundant	1 Subunit B	DI redundant
2 Subunit A	DI redundant	3 Subunit B	DI redundant

```
         Input  0.0

Signal group                          (0..31):   4
```

Bits ->	0	1	2	3	4	5	6	7

```
Discrepancy time                    (OB1/OB13):  OB1      1 cycles
Short-circ. test DQ (Y 33.0..33.3,34.0..34.3/N):  Yes  33.0
```

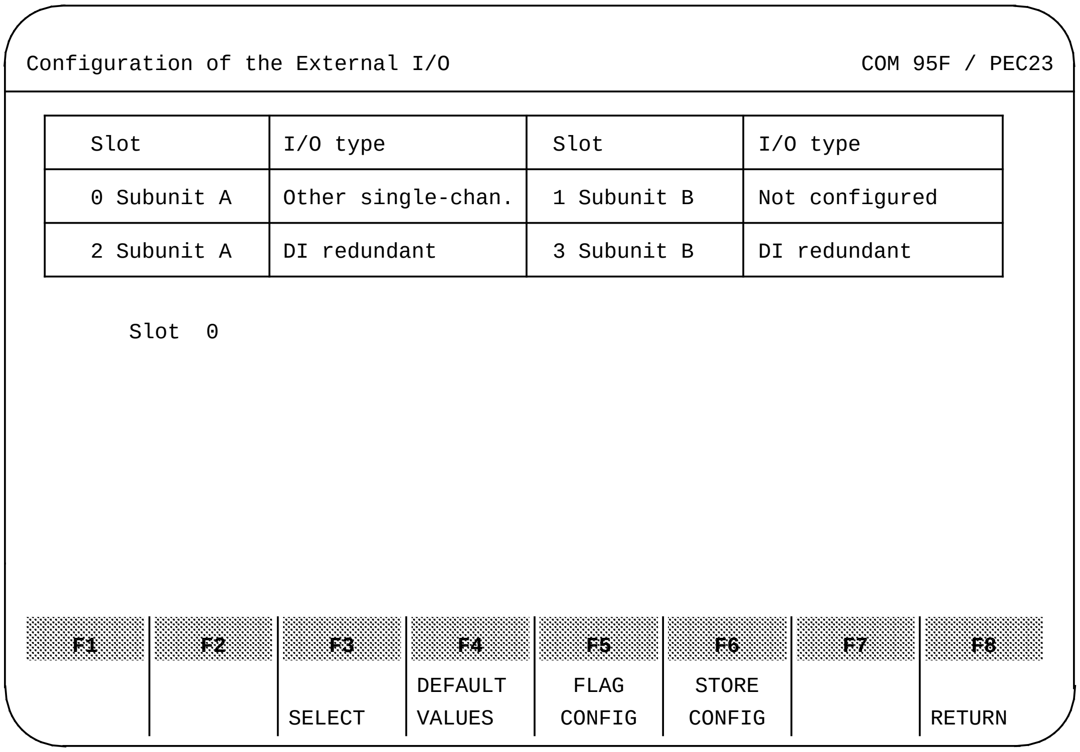
Figure 2-13. Example: Configuration of the External I/O (safety-related)

Parameterization notes

Table 2-13. Configuration of the External I/O (safety-related)

Parameters	Remarks
`Signal group`	Enter here the number of the signal group to which you want to assign I/Os. The assignment is significant for the system response to I/O failure (see Section 2.5).
`Discrepancy time`	Enter here the period for which the system is to tolerate different input values in subunit A and subunit B.
`Short-circ. test DQ`	If you want to use the line monitor function, enter here the DQ bit via which the input bit is supplied and clocked during the short-circuit test.

Example: Parameterization of non-safety-related I/O



Configuration of the External I/O COM 95F / PEC23

Slot	I/O type	Slot	I/O type
0 Subunit A	Other single-chan.	1 Subunit B	Not configured
2 Subunit A	DI redundant	3 Subunit B	DI redundant

Slot 0

Figure 2-14. Example: Configuration of the External I/O (not safety-related)

Parameterization notes

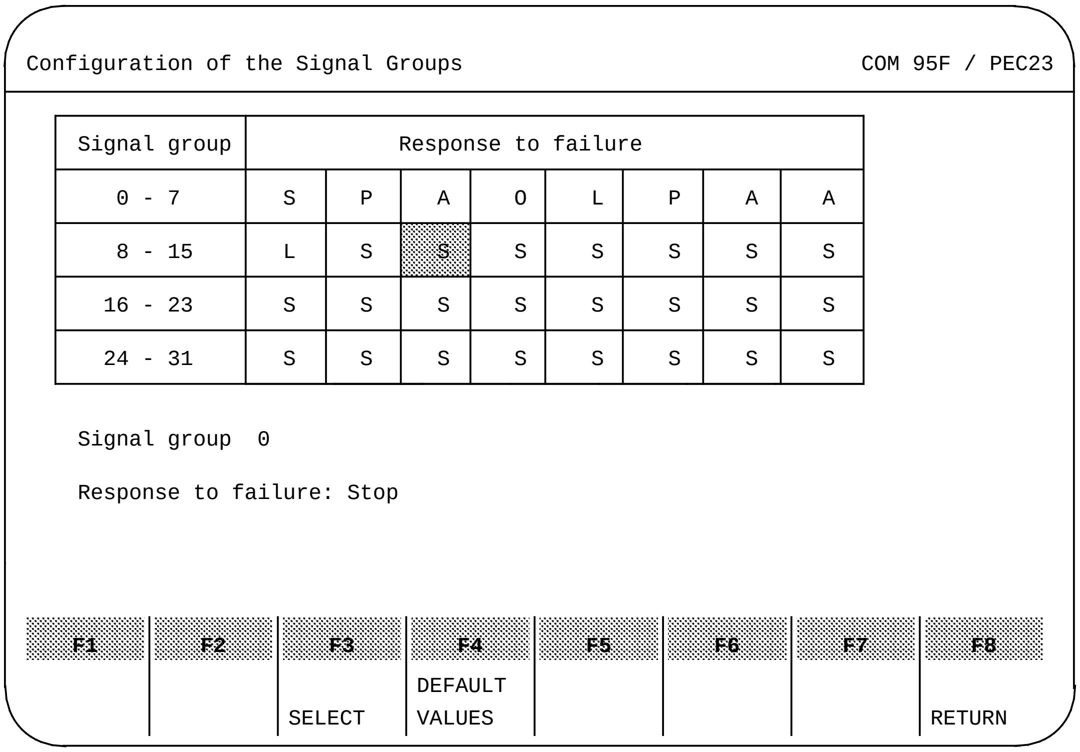
Table 2-14. Configuration of the External I/O (not safety-related)

Parameters	Remarks
`Slot 0`	Inform the S5-95F of the slot containing the non-safety-related module. Further parameters are not required.

2.5 Parameterizing the System Response to I/O Failure

Assigning signal group parameters is an essential part of system parameterization. When you parameterize the signal groups, you define the S5-95F response to a hardware failure. See also Chapter 18 of the S5-95F manual.

Example: Parameterizing the system response (signal groups)



Configuration of the Signal Groups COM 95F / PEC23

Signal group	Response to failure							
0 - 7	S	P	A	O	L	P	A	A
8 - 15	L	S	S	S	S	S	S	S
16 - 23	S	S	S	S	S	S	S	S
24 - 31	S	S	S	S	S	S	S	S

Signal group 0

Response to failure: Stop

Figure 2-15. Example: Configuration of the Signal Groups

Parameterization notes

Table 2-15. Configuration of the Signal Groups

Parameters	Remarks
Response to failure	Press <F3> to select the system response the S5-95F is to make in the event of hardware failure of the affected signal group. Legend: S = STOP P = Passivation A = AND Signals of the two differing DI bits O = OR Signals of the two differing DI bits L = Read in old value if DI bits differ

3 Evaluating the S5-95F Error Messages

Figures

3 Evaluating the S5-95F Error Messages

The specific COM 95F diagnostics functions are reached via the main menu. COM 95F offers the following functions for diagnostics and analysis of faults:

- Output of error messages as plaintext
- Output of the static error image

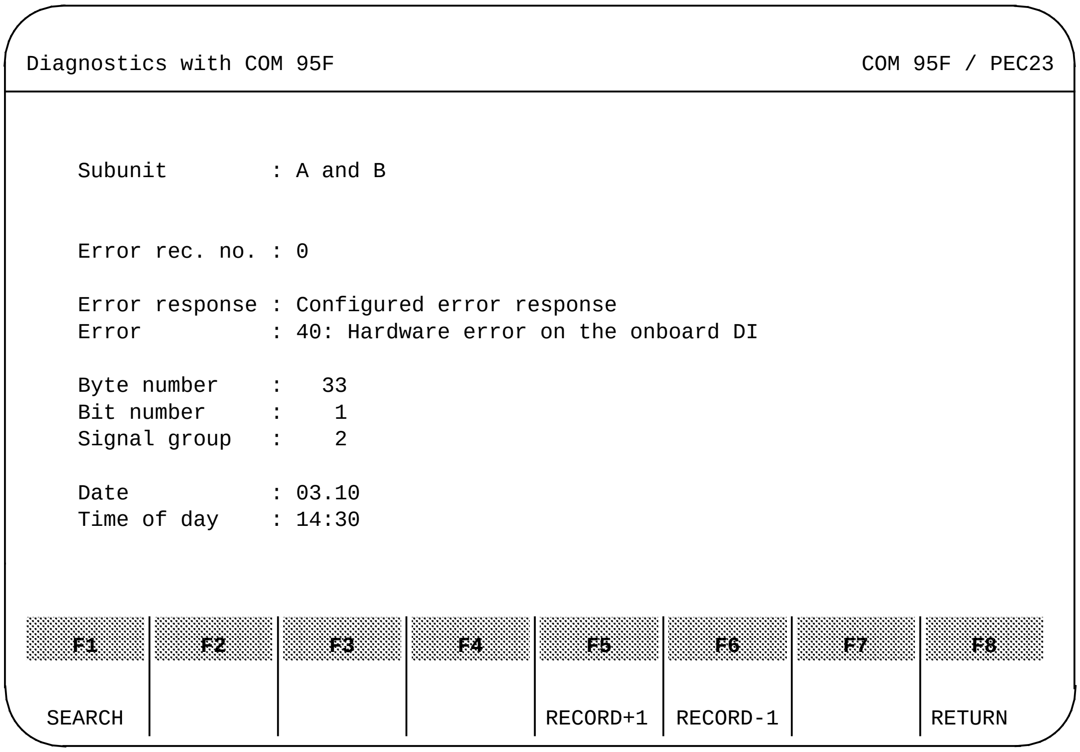
3.1 Output of Error Messages as Plaintext

All errors detected by the S5-95F are coded and entered in a system message DB. COM 95F interprets these coded entries and outputs the relevant plaintext message in each case. The following information is included in the plaintext message:

- Cause of fault
- Location of fault
- Response of the S5-95F
- Time of day and date of the message

You can, of course, use the COM 95F integral Help function here also. You then get additional information and help for remedying the fault. You will also find an overview of all error messages in Chapter 15 of the S5-95F manual.

Example: Reporting an I/O error



```
Diagnostics with COM 95F                                      COM 95F / PEC23

    Subunit        : A and B


    Error rec. no. : 0

    Error response : Configured error response
    Error          : 40: Hardware error on the onboard DI

    Byte number    :   33
    Bit number     :    1
    Signal group   :    2

    Date           : 03.10
    Time of day    : 14:30
```

Figure 3-1. Example: Reporting an I/O Error

3.2 Displaying the Static Error Image

The S5-95F also enters the information for the static error image in coded form in the system message DB. COM 95F interpretes these coded entries and displays an overview table of the passivated or errored components. Passivated components are indicated in the table displays with an asterisk (*).

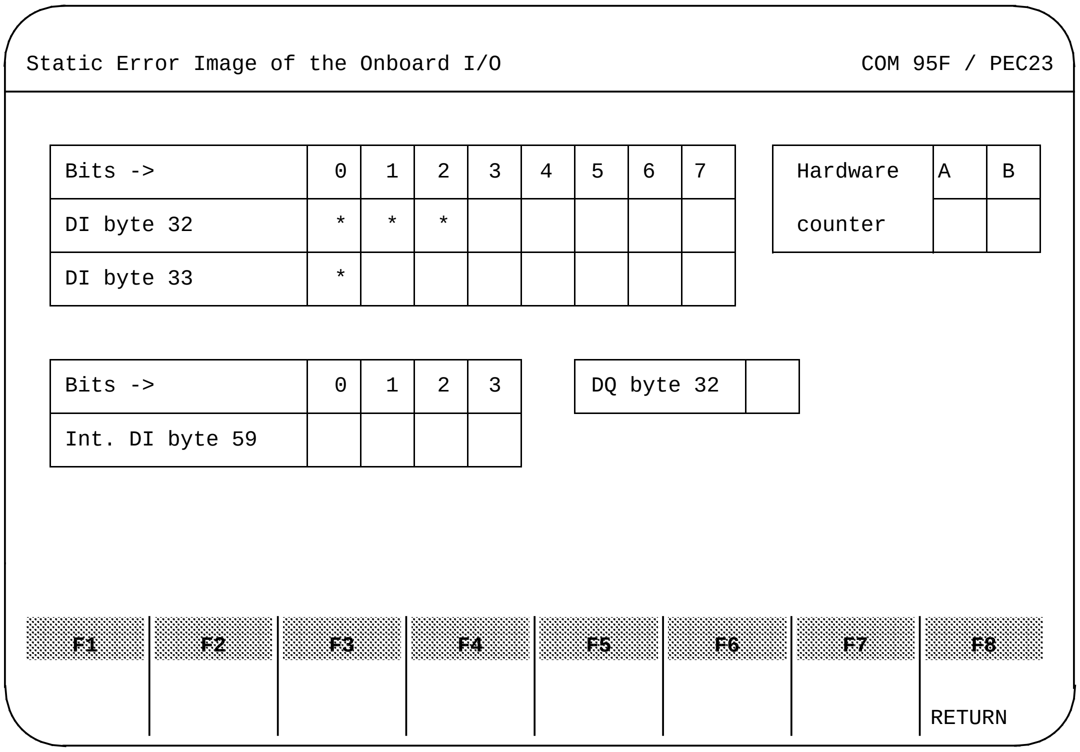
There are static error images for the following safety-related components:

- Signal groups (triggered system responses)
- Error groups
- Onboard I/O
- External I/O
- Safety-related SINEC L1 data paths

Refer to sections 15.3.4 to 15.3.7 of the S5-95F manual (Order No. 6ES5 998-1MF23) for information on resetting the static error displays.

You can, of course, use the COM 95F integral Help function here also. You then get additional information and help for remedying the fault.

Example: Static error image of the onboard I/O



Static Error Image of the Onboard I/O COM 95F / PEC23

Bits ->	0	1	2	3	4	5	6	7
DI byte 32	*	*	*					
DI byte 33	*							

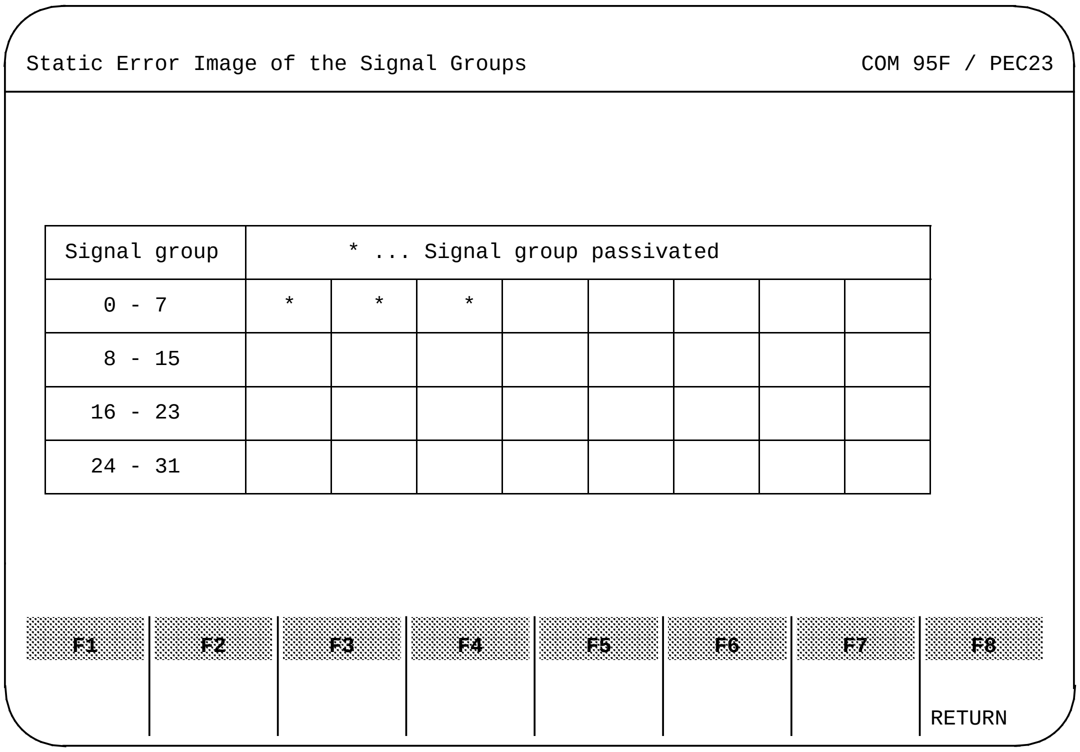
Hardware	A	B
counter		

Bits ->	0	1	2	3
Int. DI byte 59				

DQ byte 32	

Figure 3-2. Example: Static Error Image of the Onboard I/O

Example: Static error image of the signal groups

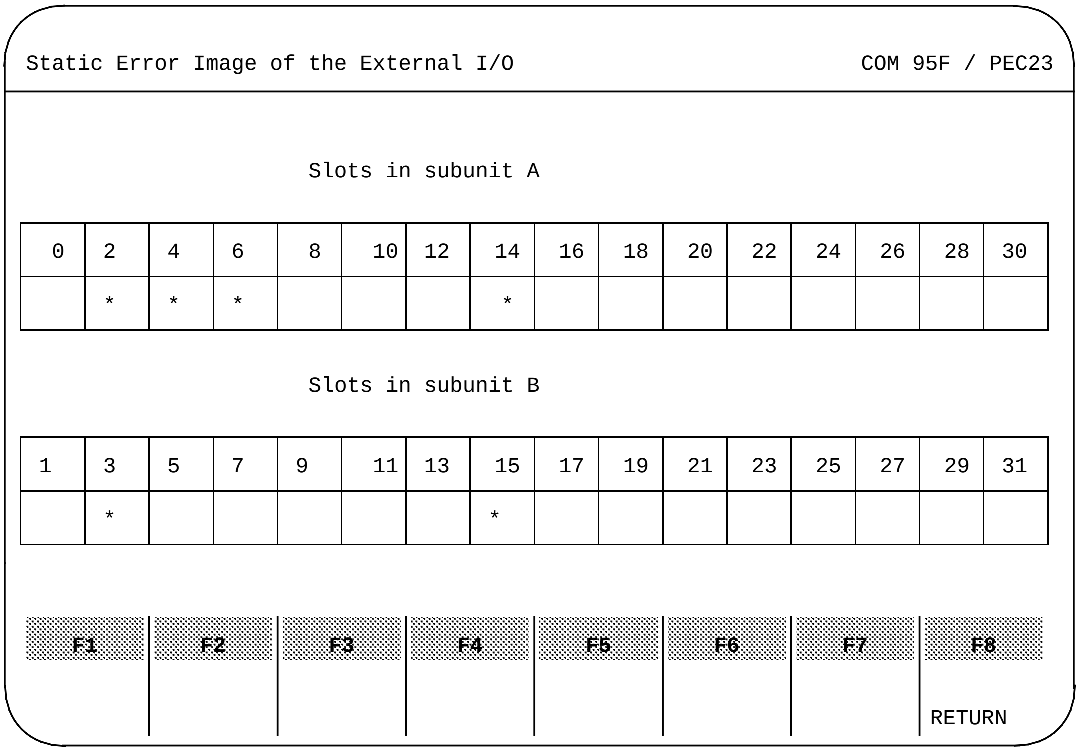


Static Error Image of the Signal Groups COM 95F / PEC23

Signal group	* ... Signal group passivated							
0 - 7	*	*	*					
8 - 15								
16 - 23								
24 - 31								

Figure 3-3. Example: Static Error Image of the Signal Groups

Example: Static error image of the external I/O



Static Error Image of the External I/O COM 95F / PEC23

Slots in subunit A

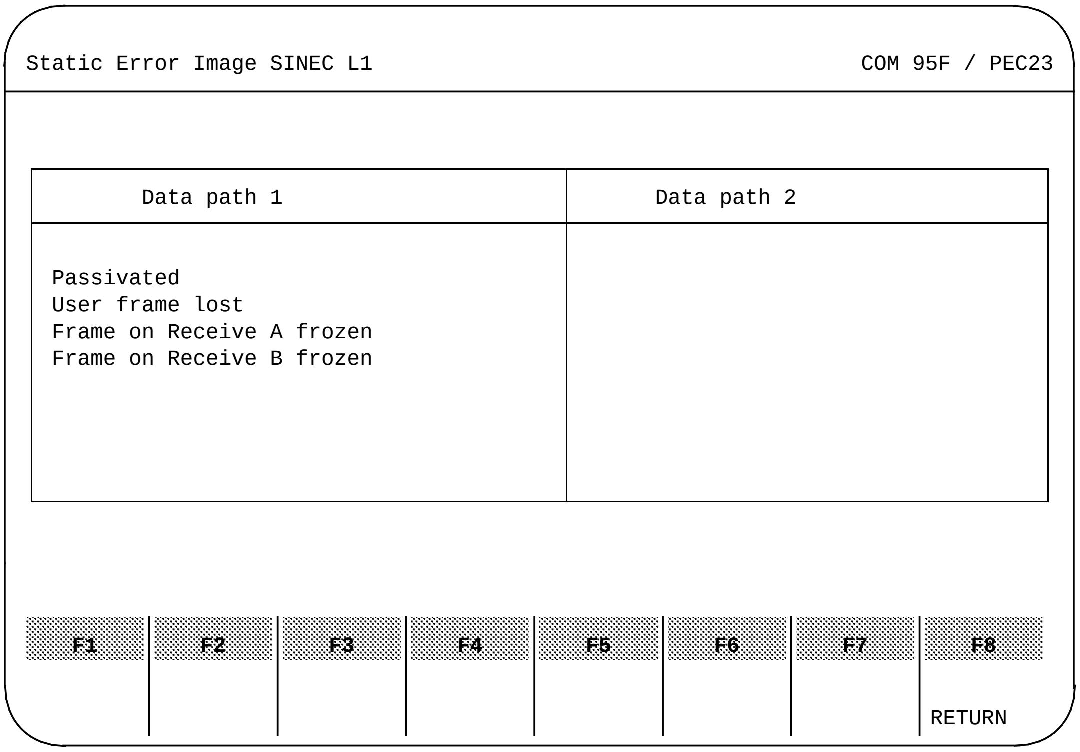
0	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30
	*	*	*				*								

Slots in subunit B

1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31
	*						*								

Figure 3-4. Example: Static Error Image of the External I/O

Example: Static error image SINEC L1



Static Error Image SINEC L1 COM 95F / PEC23

Data path 1	Data path 2
Passivated User frame lost Frame on Receive A frozen Frame on Receive B frozen	

Figure 3-5. Example: Static Error Image SINEC L1

4 Documenting the System Parameters

Table

4 Documenting the System Parameters

You require a printout of your system parameters for the system acceptance. COM 95F generates a printout specifying the program file.

Using the programmer with the STEP 5 Basic Package to V 3.x, you reach the Print menu via the System Handling menu. Using the programmer with the STEP 5 Basic Package from V 6.x, you reach the Print menu via the Documentation drop-down menu (DDM).

In the Print menu, you can specify whether you want to print the entire configuration data or only a specific area.

Example: Printing out the all system parameters

Table 4-1. Configuration Printout Example

Configuration Printout Example

```
Program file: D:JOHNFRST.S5D        CRC sum (PLC): 00000           COM 95F
================================================================================
Parameterization of the 95F Operating System                    COM 95F / PEC23
--------------------------------------------------------------------------------


  User program signature (CRC checksum):                  0


  System identification number                    (0..255):           0
  Time interval for OB 13                       (0..65535):           0 * 10ms
  DB area with constant contents              (Y 2..251/N):   Yes     2 to 251
  Maximum PLC cycle time                         (10..255):           25 * 10ms
  PLC cycle time statistics                          (Y/N):   No
  DB no. for operator input in F mode         (Y 2..251/N):   No
  Transfer of error DB via SINEC L1 (SUBUNIT B)      (Y/N):   No
  Transfer of error DB via CP 521                    (Y/N):   No




--------------------------------------------------------------------------------
Parameterization of the S5-95F System Clock                     COM 95F / PEC23
--------------------------------------------------------------------------------


  Status/control word                (Y FW,IW,QW,DB.DW /N):   No
  Clock data 22 words                (Y FW,IW,QW,DB.DW /N):   No
  Date                                     (Y dd.mm.yy /N):   No

  Enable operating hours counter                     (Y/N):   No
  Operating hours counter              (Y hhhhhh:mn:ss /N):   No
  Clock prompt date                       (Y wd dd.mm. /N):   No

  Update in STOP                                     (Y/N):   No
  Save clock time at RUN->STOP/NAU                   (Y/N):   No
  Correction factor                           (-400..+400):           0
```

Table 4-1. Configuration Printout Example (Continued)

Configuration Printout Example

```
--------------------------------------------------------------------------------
Parameterization of SINEC L1 for AG95F                           COM 95F / PDC23
--------------------------------------------------------------------------------

       PG BUS number      (Y 1..30/N):  Yes    2

       SINEC-L1 to subunit A   (Y/N) :  Yes
       SINEC-L1 to subunit B   (Y/N) :  Yes

       Slave number           (1..30):       2


--------------------------------------------------------------------------------
Parameterization of SINEC L1 in Non-Safety-Related Operation    COM 95F / PEC23
--------------------------------------------------------------------------------

     Non-safety-related data exchange with the master or other slaves


 +---------------------------------------------------+--------------------+
 ! Coordination byte send CBS        (Y FY,DB.DW /N) ! No                 !
 !---------------------------------------------------+--------------------!
 ! Coordination byte receive CBR     (Y FY,DB.DW /N) ! No                 !
 !---------------------------------------------------+--------------------!
 ! Send mailbox SM                        (FY,DB.DW) !                    !
 !---------------------------------------------------+--------------------!
 ! Receive mailbox RM                     (FY,DB.DW) !                    !
 +---------------------------------------------------+--------------------+



--------------------------------------------------------------------------------
Parameterization of SINEC L1 for Safety-Related Operation       COM 95F / PEC23
--------------------------------------------------------------------------------

     Safety-related data exchange with another S5-95F or S5-115F PLC

        Control byte (UVB) (Y 0..255/N):  No
+--------------------------------------+--------------------------------------+
!        Data path 1 (DB 252)          !        Data path 2 (DB 253)          !
!--------------------------------------+--------------------------------------!
! Send subunit (Y A,B,H/N): Yes  A     ! Send subunit (Y A,B,H/N): No         !
! Broadcast message  (Y/N): Yes        !                                      !
!                                      ! Send to slave  (0/1..30):  0         !
! Mode    (95F,115F-14/15): 115F-15    ! Mode    (95F,115F-14/15):   95F      !
! Safety time  (0/3..3276):   20*100ms ! Safety time  (0/3..3276):     0*100ms!
!--------------------------------------+--------------------------------------!
! Rec. subunit (Y A,B,H/N): No         ! Rec. subunit (Y A,B,H/N): No         !
! Rec. from slave(0/1..30):  0         ! Rec. from slave(0/1..30):  0         !
! Mode    (95F,115F-14/15):   95F      ! Mode    (95F,115F-14/15):   95F      !
! Safety time  (0/3..3276):     0*100ms! Safety time  (0/3..3276):     0*100ms!
! Response     (Stop,user): Stop       ! Response     (Stop,user): Stop       !
+--------------------------------------+--------------------------------------+
```

Table 4-1. Configuration Printout Example (Continued)

Configuration Printout Example

```
--------------------------------------------------------------------------------
Configuration of the Onboard Counters                           COM 95F / PEC23
--------------------------------------------------------------------------------

                    Counters cascaded:  No


                ------------------------------------------------
                Counter A activated      (Y/N):  No
                ---------
                Signal group           (0..31):   0
                Counting at edge     (pos/neg):  negative edge
                Comparison value    (0..65535):      0
                ------------------------------------------------
                Counter B activated      (Y/N):  No
                ---------
                Signal group           (0..31):   0
                Counting at edge     (pos/neg):  negative edge
                Comparison value    (0..65535):      0
                ------------------------------------------------




--------------------------------------------------------------------------------
Configuration of the Onboard DQs (DQ 32)                        COM 95F / PEC23
--------------------------------------------------------------------------------


  ================!=================
  !  DQ byte 32   !  DQ redundant  !
  ================!=================


 Digital output byte 32

 Signal group (0..31):  0
```

Table 4-1. Configuration Printout Example (Continued)

Configuration Printout Example

```
--------------------------------------------------------------------------------
Configuration of the Onboard DIs (DI 32/33)                     COM 95F / PEC23
--------------------------------------------------------------------------------
        Redundant byte:   DI 32             Signal group:  0
--------------------------------------------------------------------------------
 Onboard    Short             OB3 interrupt             Discrepancy      Sh. circ.
 bit        symbol                                      time             test
-------------------------------------------------------------------------------
! 32.0  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 32.1  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 32.2  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 32.3  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 32.4  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 32.5  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 32.6  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 32.7  !            !  No                          ! short  1ms        !      !
+-------+------------+------------------------------+-------------------+------+
        Redundant byte:  DI 33               Signal group:  0
--------------------------------------------------------------------------------
! 33.0  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 33.1  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 33.2  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 33.3  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 33.4  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 33.5  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 33.6  !            !  No                          ! short  1ms        !      !
!-------+------------+------------------------------+-------------------+------!
! 33.7  !            !  No                          ! short  1ms        !      !
+-------+------------+------------------------------+-------------------+------+
--------------------------------------------------------------------------------
Configuration of the Onboard DIs (DI 59.0..59.3)                COM 95F / PEC23
--------------------------------------------------------------------------------
        Redundant byte:  DI 59               Signal group:  0
--------------------------------------------------------------------------------
 Onboard    Short             OB2 interrupt             Discrepancy      Sh. circ.
 bit        symbol                                      time             test
-------------------------------------------------------------------------------
! 59.0  !            ! OB2 interrupt                ! 		            !      !
!-------+------------+------------------------------+-------------------+------!
! 59.1  !            ! OB2 interrupt                !                   !      !
!-------+------------+------------------------------+-------------------+------!
! 59.2  !            ! OB2 interrupt                !                   !      !
!-------+------------+------------------------------+-------------------+------!
! 59.3  !            ! OB2 interrupt                !                   !      !
+-------+------------+------------------------------+-------------------+------+
```

Table 4-1. Configuration Printout Example (Continued)

Configuration Printout Example

```
---------------------------------------------------------------------------------
Configuration of the External I/O                               COM 95F / PEC23
---------------------------------------------------------------------------------
Slot   !Short   !I/O type          !Signal
       !symbol  !                  !group
---------------------------------------------------------------------------------
!IB 0  !        ! DI redundant     !  4                                       !
+------+--------+------------------+---------------------------------------+
 Bit!Short   !Discrepancy     !Short circuit
    !symbol  !        cycles  !test
+---+--------+----------------+------------
! 0 !        !OB1      1      !33.0
! 1 !        !OB1      1      !33.0
! 2 !        !OB1      1      !
! 3 !        !OB1      1      !
! 4 !        !OB1      1      !
! 5 !        !OB1      1      !
! 6 !        !OB1      1      !
! 7 !        !OB1      1      !
-------------------------------------------------------------------------------
!IB 2  !        ! DI redundant     !  3                                       !
+------+--------+------------------+---------------------------------------+
 Bit!Short   !Discrepancy     !Short circuit
    !symbol  !        cycles  !test
+---+--------+----------------+------------
! 0 !        !OB1      1      !
! 1 !        !OB1      1      !
! 2 !        !OB1      1      !
! 3 !        !OB1      1      !
! 4 !        !OB1      1      !
! 5 !        !OB1      1      !
! 6 !        !OB1      1      !
! 7 !        !OB1      1      !
-------------------------------------------------------------------------------
!IB 4  !        ! DI redundant     !  0                                       !
+------+--------+------------------+---------------------------------------+
 Bit!Short   !Discrepancy     !Short circuit
    !symbol  !        cycles  !test
+---+--------+----------------+------------
! 0 !        !OB1      1      !
! 1 !        !OB1      1      !
! 2 !        !OB1      1      !
! 3 !        !OB1      1      !
! 4 !        !OB1      1      !
! 5 !        !OB1      1      !
! 6 !        !OB1      1      !
! 7 !        !OB1      1      !
-------------------------------------------------------------------------------
!   6  !        ! Not configured   !  0                                       !
!------+--------+------------------+---------------------------------------!
!   7  !        ! Not configured   !  0                                       !
!------+--------+------------------+---------------------------------------!
```

Table 4-1. Configuration Printout Example (Continued)

Configuration Printout Example

```
Slot   !Short   ! I/O type         !Signal
       !symbol  !                  !group
---------------------------------------------------------------------------------
!   8  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!   9  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  10  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  11  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  12  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  13  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  14  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  15  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  16  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  17  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  18  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  19  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  20  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  21  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  22  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  23  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  24  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  25  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  26  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  27  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  28  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  29  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  30  !        ! Not configured   !  0                                       !
!------+--------+------------------+------------------------------------------!
!  31  !        ! Not configured   !  0                                       !
+------+--------+------------------+------------------------------------------+
```

Table 4-1. Configuration Printout Example (Continued)

Configuration Printout Example

```
--------------------------------------------------------------------------------
Configuration of the Signal Groups                              COM 95F / PEC23
--------------------------------------------------------------------------------
Signal group       ! Response to failure

!    0             ! Stop                                     !
!------------------+---------------------------------------!
!    1             ! Stop                                     !
!------------------+---------------------------------------!
!    2             ! Stop                                     !
!------------------+---------------------------------------!
!    3             ! Stop                                     !
!------------------+---------------------------------------!
!    4             ! Stop                                     !
!------------------+---------------------------------------!
!    5             ! Stop                                     !
!------------------+---------------------------------------!
!    6             ! Stop                                     !
!------------------+---------------------------------------!
!    7             ! Stop                                     !
!------------------+---------------------------------------!
!    8             ! Stop                                     !
!------------------+---------------------------------------!
!    9             ! Stop                                     !
!------------------+---------------------------------------!
!   10             ! Stop                                     !
!------------------+---------------------------------------!
!   11             ! Stop                                     !
!------------------+---------------------------------------!
!   12             ! Stop                                     !
!------------------+---------------------------------------!
!   13             ! Stop                                     !
!------------------+---------------------------------------!
!   14             ! Stop                                     !
!------------------+---------------------------------------!
!   15             ! Stop                                     !
!------------------+---------------------------------------!
!   16             ! Stop                                     !
!------------------+---------------------------------------!
!   17             ! Stop                                     !
!------------------+---------------------------------------!
!   18             ! Stop                                     !
!------------------+---------------------------------------!
!   19             ! Stop                                     !
!------------------+---------------------------------------!
!   20             ! Stop                                     !
!------------------+---------------------------------------!
!   21             ! Stop                                     !
!------------------+---------------------------------------!
!   22             ! Stop                                     !
!------------------+---------------------------------------!
!   23             ! Stop                                     !
!------------------+---------------------------------------!
```

Table 4-1. Configuration Printout Example (Continued)

Configuration Printout Example

```
Signal group       ! Response to failure

!   24             ! Stop                                  !
!------------------+---------------------------------------!
!   25             ! Stop                                  !
!------------------+---------------------------------------!
!   26             ! Stop                                  !
!------------------+---------------------------------------!
!   27             ! Stop                                  !
!------------------+---------------------------------------!
!   28             ! Stop                                  !
!------------------+---------------------------------------!
!   29             ! Stop                                  !
!------------------+---------------------------------------!
!   30             ! Stop                                  !
!------------------+---------------------------------------!
!   31             ! Stop                                  !
+------------------+---------------------------------------+
```

5 Acceptance Tool

Figure

Tables

5 Acceptance Tool

In addition to the configuring software, the COM 95F software package also contains a tool (called the acceptance tool) which simplifies approval of the failsafe system.

The acceptance tool consists of the following:

- A change comparator (revision comparator) to document modified user programs
- A logical comparator to detect data corruption in the user programs resulting from systematic errors and/or hardware failure in the programmer.

Change comparator/Revision comparator

The change/revision comparator makes it particularly easy for you to show the differences between two user programs.

The comparator is optional and always serves a practical purpose when the inspector has already examined the program once before. Should a re-examination be needed, it is often necessary for the inspector to examine only the program sequences which have been modified.

Logical comparator

The logical comparator is used to prove the coincidence of two user programs. The purpose of the comparator is to detect changes in the user program caused by ***hardware failures in the programmer*** (not very probable) or ***systematic errors in the programmer software***.

The use of the logical comparator is necessary if a complete functional test of the plant is not carried out during the acceptance test.

To check the user program for changes, you need two ASCII files which have been generated completely separately from each other. Sections 5.2.1 and 5.2.2 show how this is done.

5.1 Installing the Acceptance Tool on the Hard Disk

This section shows you how to install the acceptance tool on your programmer/PC. We assume that the floppy drive is defined as A: and the hard disk as C:.

Installing the Acceptance Tool on a Programmer/PC with STEP 5 Basic Package up to Version V 6.x

If you are using STEP 5 basic package version V 3.x or V 6.x, proceed as described in the following table to install the tool on the hard disk.

Prerequisite for installation on a PC:
If you are using a PC, the STEP 5 basic package must be stored in the `STEP 5/S5-ST or SIMATIC/S5_ST` directory on drive C:\, D:\ or E:\.

Table 5-1. Installing Acceptance Tool on Programmer/PC with STEP 5 Basic Package up to Version V 6.x

Step	Action	Response
1	Insert the disk in drive A:	
2	Select drive A: and call up the installation program `INST_F.EXE`	The opening menu of the installation program appears on the monitor
3	Select the "Acceptance Tool" package in the installation selection and answer the questions on the screen	The parameterization software COM 95F is automatically installed. When the installation is complete, the following message box appears on the screen: "Acceptance Tool successfully installed"
4	Restart the programmer	The system accepts the relevant variables for installation of the tool

Special points for installation on a programmer/PC with MS-DOS version lower than V 5.0:
If your programmer is not yet equipped with MS-DOS V 5.0, please install the Acceptance Tool in the "ABN" directory suggested by the installation program. Furthermore, your user programs must be located on hard disk C:.

Install the acceptance tool as described in Steps 1 - 3
Copy the `ABN_B_CO.BAX` file with

```
Copy ABN_B_CO.BAX ABN_B_CO.BAT
```

Restart the programmer.

If you want to install the Acceptance Tool in a directory other than "ABN", you must modify the `ABN_B_CO.BAX` file before copying.

Note

The MS-DOS program `share.exe` **must not be started** when you are working with the Acceptance Tool. Please check in the `autoexec.bat` file to ensure that the `share.exe` program is not started there.

Installing the Acceptance Tool on a Programmer/PC with Step 5 Basic Package Version V 7.0 Upwards

If your programmer/PC is equipped with a STEP 5 basic package version V 7.0 upwards, proceed with the installation as described in the table below.

Table 5-2. Installing Acceptance Tool on Programmer/PC with STEP 5 Basic Package Version V 7.0 Upwards

Step	Action	Response
1	Insert the disk in drive A:	
2	Select drive A: and call up the installation program `INST_F.EXE`	The opening menu of the installation program appears on the monitor
3	Select the "Acceptance Tool" package in the installation selection and answer the questions on the screen	The parameterization software COM 95F is automatically installed. When the installation is complete, the following message box appears on the screen: "Acceptance Tool successfully installed"
4	Restart the programmer	The system accepts the relevant variables for installation of the tool

5.2 Conventions

To make it easier to read, we will call the file with your STEP 5 user program `FILE01ST.S5D`. You must generae further files from this file for both comparators.

The generated files are the same except for the file name extension. Table 5-3 gives you an initial overview of the user files and is designed to familiarize you with the writing convention.

Table 5-3. Overview of the User Files

File Name	Task of the File
`FILE01ST.S5D`	Contains the STEP 5 user program.
`FILE01A0.SEQ`	Contains STL-ASCII file (generated with BATCH compiler).
`FILE01LS.INI`	Contains STL-ASCII file (generated with KOMDOK).
`DELTA.MUP`	Contains the change comparator (revision comparator) protocol.
`STL_COMP.LOG`	Contains the logical comparator protocol.

5.2.1 How to Generate an STL-ASCII File for the Change Comparator and the Logical Comparator Using the BATCH Compiler

Before starting the change comparator or the logical comparator, you must generate an STL-ASCII file (`FILE01A0.SEQ`) with the help of the BATCH compiler from your STEP 5 user program (`FILE01ST.S5D`). The BATCH compiler is a component of the acceptance tool.

Table 5-4 shows how you generate an STL-ASCII file with the BATCH compiler under MS-DOS.

Table 5-4. Generating an STL-ASCII File with the BATCH Compiler under MS-DOS

Step	Action
1	Call the BATCH compiler from the command line level. `ABN_B_CO FILE01` *The* `ST.S5D` *extension is no longer specified.*
2	If the system detects an error after the first compiler step, you will be asked if you wish to continue with compiling: • Continue compiling if the errors shown on screen are not significant, e.g. "Pre-header of a DB missing" • Terminate compiling if serious errors are reported, e.g. "Opcode error". The file with the name `FILE01AF.SEQ` contains the list of errors for such cases.

Table 5-5 shows how you generate an STL-ASCII file with the BATCH compiler under PCP/M.

Table 5-5. Generating an STL-ASCII File with the BATCH Compiler under PCP/M

Step	Action
1	Call the BATCH compiler from the command line level. `ABN_B_CO FILE01ST.S5D FILE01A0.SEQ E`
2	If the system detects an error after the first compiler step, you will be asked if you wish to continue with compiling: • Continue compiling if the errors shown on screen are not significant, e.g. "Pre-header of a DB missing" • Terminate compiling if serious errors are reported, e.g. "Opcode error". The file with the name `FILE01AF.SEQ` contains the list of errors for such cases.

5.2.2 How to Generate an STL-ASCII File for the Logical Comparator Using the KOMDOK Software Package

Before starting the logical comparator, you must generate an STL-ASCII file (`FILE01LS.INI`) from your STEP 5 user program (`FILE01ST.S5D`) with the help of the KOMDOK software package. The KOMDOK software package is a component of the STEP 5 Basic Package from Version 6.0. If you have a Basic Package with an earlier update you must use the KOMDOK software package in addition to this.

Table 5-6 shows how you generate an STL-ASCII file with KOMDOK.

Table 5-6. Generating an STL-ASCII File Using the STEP 5 Basic Package to V 3.x and Separate KOMDOK Software Package

Step	Action
1	Make sure you have entered a wide footer (132 characters) in a footer file. See your programmer manual for details of how to create a footer file.

2	Direct the printer output to a file.
	Directing printer output to a file is described in detail in your programmer manual.
	Set the following when directing the printer output to file:
	`-   Printer type:              REMOTE (DIN A4)`
	`-   Output to:                 FILE01LS.INI`

3	Select the "KOMDOK" package or "Enhanced output" (depending on the STEP 5 Basic Package used).

4	Select the following in the Defaults screen form:
	`-   Method of representation:    STL`
	`-   Symbols:                     No`
	`-   Footer file:                 FILE01F2.INI`

Note

Make sure that you write the documentation to an ***empty*** print file. Otherwise, the documentation will be appended to the previously output file and cause errored comparisons.

The enhanced documentation function (KOMDOK) is already integrated in the STEP 5 Basic Package from V 6.0. Table 5-7 shows how you generate an STL-ASCII file.

Table 5-7. Generating an STL-ASCII File with STEP 5 Basic Package from V 6.0 and Integrated Documentation Function

Step	Action
1	Make sure you have entered a wide footer (132 characters) in a footer file (e.g. `FILE01F2.INI`). See your programmer manual for details of how to create a footer file.
2	Select the "STEP 5" package
3	Select the following on page 2 of the Settings screen form: - `Method of representation:` `STL` - `Symbols:` `No` - `Output to file:` `FILE01L3.INI` - `Footer:` `132 characters`
4	Select the following on page 1 of the Settings screen form: - `Program file:` `FILE01ST.S5D` - `Footer file:` `FILE01F2.INI`
5	Select "Enhance output" menu in the "Documentation" menu and activate the next menu "Program sections" incl. "Blocks".
6	Acknowledge the dialog box with the standard setting.

Note

Make sure that you write the documentation to an ***empty*** print file. Otherwise, the documentation will be appended to the previously output file and cause errored comparisons.

5.2.3 How to Generate a KOMDOK Listing for the Pre-Inspection

The inspector requires a KOMDOK listing of your user program for the pre-inspection. You can output the user program in LAD, CSF or STL.

To achieve consistency between the listing printed later with KOMDOK and the program file, generate a check file from your program file `FILE01ST.S5D` using COM 95F. COM 95F generates this check file autonomously.

Generating a check file with COM 95F

COM 95F copies your STEP 5 program file to a second file and generates a checksum. The checksum becomes a component of the file name of the second file (check file).

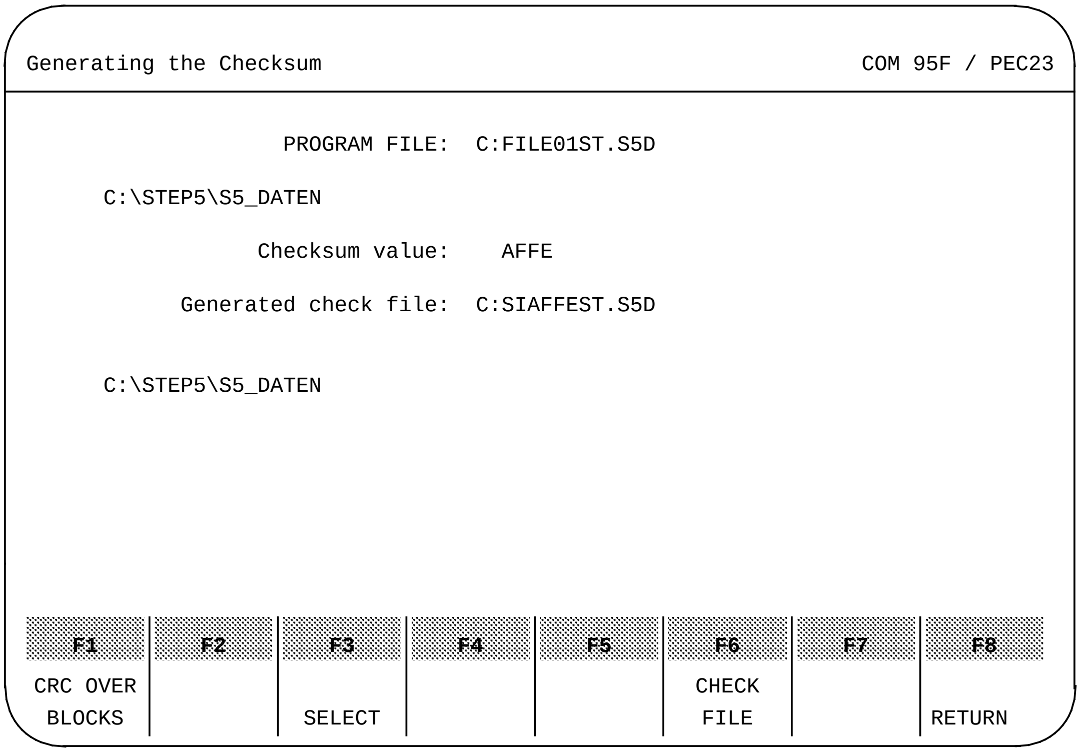
Check files always start with the letters "SI" followed by four characters for the checksum. The full name of a check file is `SIxxxx.ST.S5D` where `xxxx` represents the checksum calculated. Since the checksum is a component of the file name, it appears on every printed page; you can then see easily which printout belongs to which user program software.

The check file automatically receives the attribute "read only" and so is protected against inadvertant changes.

Table 5-8. Generating a Check File with COM 95F

Step	Action	Response
1	Select the "Generate CRC" menu in the COM 95F software package	The "Generate checksum" screen form appears
2	Enter your file with the user program in the screen form and press <F6> "Start".	COM 95F generates the checksum and copies the check file with the user program into the directory. The file is called `SIxxxxST.S5D`.

Example: Generating the checksum



```
Generating the Checksum                                          COM 95F / PEC23

                    PROGRAM FILE:  C:FILE01ST.S5D

      C:\STEP5\S5_DATEN

                  Checksum value:    AFFE

            Generated check file:  C:SIAFFEST.S5D


      C:\STEP5\S5_DATEN
```

Figure 5-1. Example: Generating the Checksum

Printing out the KOMDOK listing for the inspector

The inspector requires a user program listing for checking the user program. Before printing out the check file generated with COM 95F (`SIxxxxST.S5D`) via KOMDOK, you must generate a footer with a width of 132 characters. Generation of this footer file is described in detail in your programmer manual. The tables below represent only a brief account of this procedure.

If you have a STEP 5 Basic Package to V 3.x, you require the KOMDOK software package additionally.

Table 5-9. Printing Out the KOMDOK Listing for the Inspector with the STEP 5 Basic Package to V 3.x and the KOMDOK Package

Step	Action
1	Make sure you have entered a wide footer (132 characters) in a footer file. See your programmer manual for details of how to create a footer file.
2	Select the "KOMDOK" package
3	Select the following in the Defaults screen form: - Program file: SIAFFE.S5D - Method of representation: LAD, CSF or STL - Footer file: FILE01F2.INI
5	Select the "Block" menu in the "Document" menu.
6	Select "B" to output all blocks and confirm the entry by pressing the Enter key.

If you have a STEP 5 Basic Package from V 6.0, the KOMDOK software package is already integrated.

Table 5-10. Printing Out the KOMDOK Listing for the Inspector with the STEP 5 Basic Package from V 6.0

Step	Action
1	Make sure you have entered a wide footer (132 characters) in a footer file. See your programmer manual for details of how to create a footer file.
2	Select the "STEP 5" package
3	Select the following on page 2 of the Settings screen form: - Method of representation: STL - Footer: 132 characters
4	Select the following on page 1 of the Settings screen form: - Program file: SIAFFEST.S5D - Footer file: FILE01F2.INI
5	Select the "Enhanced output" menu in the "Documentation" menu and activate the next menu "Program sections" followed by "Blocks".
6	Acknowledge the dialog box with the standard setting.

5.3 Using the Change Comparator (Revision Comparator)

The change comparator (revision comparator) has the task of detecting differences between two program files. You can compare complete files or individual blocks using the change comparator (revision comparator). The comparator requires two STL-ASCII files which you have generated from the two program files using the BATCH compiler (see Section 5.3.1).

The change comparator (revision comparator) is to be carried out at the following points:

- At the start of the system acceptance.
  The check file from the pre-inspection is to be compared with the program file from the (first) system version.

- At the end of the system acceptance.
  The first system version is to be compared with the final (accepted) system version.

Both change comparator (revision comparator) are required in order to detect changes in the user program caused by dynamic hardware failures in the programmer which can occur during the system acceptance.

The change comparator (revision comparator) detects any changed program sequences and enters them in the file DELTA.MUP. You can output the detected changes to printer or screen and then evaluate them with the inspector.

The change comparator (revision comparator) is also useful in cases where the user program has already been presented to the inspector. When a test is being repeated, it is frequently only necessary to evaluate the changed program sequences.

5.3.1 Working with the Change Comparator (Revision Comparator)

For the change comparator (revision comparator), you require two STL-ASCII files generated from the two program files (e.g. `FILE01A0.SEQ` and `FILE02A0.SEQ`) with the help of the BATCH compiler.

Table 5-11 shows how you execute the change comparator (revision comparison).

Table 5-11. Calling the Change Comparator (Revision Comparator)

Step	Action	Response
1	Start the installed acceptance tool with `ABN_F`	The acceptance tool appears with the Start screen form.
2	Select the change comparator (revision comparator) with <F1>.	The input screen form of the change comparator (revision comparator) appears.
3	Enter the file names of the STL-ASCII files (extension `A0.SEQ`) in the two fields and confirm the entry.	The selection form for "Full comparison/block comparison" appears.
4	Specifiy whether you want to execute a file comparison or a block comparison and confirm your selection.	The comparator checks the two files and writes the changed program sequences into the `DELTA.MUP` file. It then asks the system whether the changes are to be printed out or displayed on screen.

5.3.2 Evaluating the Change (Revision) Comparison Protocol

This section contains examples designed to show you how to evaluate and interpret detected changes.

The change comparator (revision comparator) protocols the detected differences in "Deviation blocks". The comparator

- begins with an entry as soon as it detects different program lines
  and
- ends the entry only after it has detected at least two identical program lines.

An output deviation block consists of the following:

- Last program line still identical in both files
- All divergent program lines
- The first of two contiguous program lines which are identical in both files.

Note

Unexplainable deviations in the change comparator (revision comparator), i.e. deviations not programmed by the programmer, indicate an MC5 code corruption as a result of programmer software errors or programmer hardware failures.
Load the errored blocks into the S5-95F and find out the absolute address of the errored statements. Compare the entered MC5 code wiht the code of the programmed operation (see S5-95F manual, Appendix C).

Change comparator (revision comparator) printout examples

Table 5-12. Example 1: Replacing Lines

Example 1: Replacing lines

In the first deviation block, the comparator has detected that the statement "A I 32.1" in FB1, Segment 1, has been replaced by the statement "A I 32.2":

```
***** C:FILE01A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    3:                A   I  32.0
    4:                A   I  32.1
    5:                =   Q  32.0
***** C:FILE02A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    3:                A   I  32.0
    4:                A   I  32.2
    5:                =   Q  32.0

**********************************************************************
```

In the second deviation block, the comparator has detected that the statement "T FW 1" in FB1, Segment 1 has been replaced by the statement "T FW 0":

```
***** C:FILE01A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    7:                L   KH 1234
    8:                T   FW 1
    9:                BE
***** C:FILE02A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    7:                L   KH 1234
    8:                T   FW 0
    9:                BE

**********************************************************************
```

In the third deviation block, the comparator has detected that data word 3 in DB10 has been changed from the value "3333" to the value "3003":

```
***** C:FILE01A0.SEQ
      => ***** Block DB 10, Data word no. 2 *****

   14:                KH     2222
   15:                KH     3333
   16:                KH     4444
***** C:FILE02A0.SEQ
      => ***** Block DB 10, Data word no. 2 *****

   14:                KH     2222
   15:                KH     3003
   16:                KH     4444

**********************************************************************
```

Table 5-13. Example 2: Deleting Lines

Example 2: Deleting lines

```
In the first deviation block, the comparator has detected that the
statement "A I 32.1" in FB1, Segment 1, has been deleted.

***** C:FILE01A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    3:                A   I  32.0
    4:                A   I  32.1
    5:                =   Q  32.0
***** C:FILE03A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    3:                A   I  32.0
    4:                =   Q  32.0

**********************************************************************


In the second deviation block, the comparator has detected that the
statement "T FW 1" in FB1, Segment 1 has been deleted:

***** C:FILE01A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    7:                L   KH 1234
    8:                T   FW 1
    9:                BE
***** C:FILE03A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    6:                L   KH 1234
    7:                BE

**********************************************************************


In the third deviation block, the comparator has detected that the
original data word 3 in DB10 has been deleted:

***** C:FILE01A0.SEQ
      => ***** Block DB 10, Data word no. 2 *****

   14:                KH     2222
   15:                KH     3333
   16:                KH     4444
***** C:FILE03A0.SEQ
      => ***** Block DB 10, Data word no. 2 *****

   12:                KH     2222
   13:                KH     4444

**********************************************************************
```

Table 5-14. Example 3: Inserting Lines

Example 2: Inserting lines

In the first deviation block, the comparator has detected that the statement "A I 32.2" in FB1, Segment 1, has been inserted.

```
***** C:FILE01A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    4:                A   I  32.1
    5:                =   Q  32.0
***** C:FILE04A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    4:                A   I  32.1
    5:                A   I  32.2
    6:                =   Q  32.0

**********************************************************************
```

In the second deviation block, the comparator has detected that the statement "T FW 3" in FB1, Segment 1 has been inserted:

```
***** C:FILE01A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    8:                T   FW 1
    9:                BE
***** C:FILE04A0.SEQ
      => ***** Block FB 1, Segment no. 1 *****

    9:                T   FW 1
   10:                T   FW 3
   11:                BE

**********************************************************************
```

In the third deviation block, the comparator has detected that a new data word with the value "FFFF" has been inserted after DW3 in DB10:

```
***** C:FILE01A0.SEQ
      => ***** Block DB 10, Data word no. 3 *****

   15:                KH     3333
   16:                KH     4444
***** C:FILE04A0.SEQ
      => ***** Block DB 10, Data word no. 3 *****

   17:                KH     3333
   18:                KH     FFFF
   19:                KH     4444

**********************************************************************
```

5.4 Using the Logical Comparator

The logical comparator has the task of detecting logical corruptions in the user program caused by systematic errors and/or hardware failures in the programmer. Undetected errors could cause the compiler to store the wrong MC5 code even though the STEP 5 operation appears correctly on the printer/screen. For this reason, diverse decompilation of the program file followed by a comparison is absolutely necessary.

The logical comparator requires the following for this check:

- An STL-ASCII file generated using the BATCH compiler (see Section 5.2.1)
  and
- An STL-ASCII file generated using the KOMDOK software package (see Section 5.2.2).
  The STL-ASCII file may not contain any additional control characters. Please use the default printer file `NONAMEDR.INI` for printing in KOMDOK.

Both files must be generated from the same STEP 5 file.

> **Safety Note**
>
> If you do not carry out a complete functional test, it is imperative for safety reasons that you carry out a logical comparision before programming the EPROM module. The function of the logical comparison is to detect dangerous systematic errors and hardware failures in the programmer (the logical comparison must be complete without any errors before the EPROM module is programmed).

5.4.1 Working with the Logical Comparator

You require two STL-ASCII files for the logical comparison; one generated using the BATCH compiler (e.g. `FILE01A0.SEQ`) and a second generated using the KOMDOK software package (e.g. `FILE01LS.INI`). The method of representation must be STL.

Table 5-15. Calling the Logical Comparator

Step	Action	Response
1	Start the installed acceptance tool with `C:ABN_F`	The acceptance tool appears with the Start screen form.
2	Select the logical comparator with <F2>.	The input screen form of the logical comparator appears.
3	Enter the two input files in the two fields (one file has the extension `A0.SEQ`, and the other, the extension `LS.INI`) and confirm the entry.	The comparator compares both files and stores the protocol in the file `STL_COMP.LOG`. The system then asks you whether you want to print out the changes or display them on screen.

5.4.2 Evaluating the Logical Comparison Protocol

The logical comparison must not detect any code corruptions. A corruption message indicating unused jump labels can be ignored.

If the logical comparator detects code corruptions, load the errored blocks into the S5-95F. Compare the entered MC5 code with the code of the programmed operation (see S5-95F manual, Appendix C).

Please ensure that you print KOMDOK to an empty file.

Safety Note

If you do not carry out a complete functional test, it is imperative for safety reasons that you carry out a logical comparison before programming the EPROM module. The function of the logical comparison is to detect dangerous systematic errors and hardware failures in the programmer (the logical comparision must be complete without any errors before the EPROM module is programmed).

Index

Index

Siemens AG
A&D AS E 48
Postfach 1963

D-92209 Amberg
Federal Republic of Germany

From:

Your Name: ______________________________

Your Title: ______________________________

Company Name: ______________________________

Street: ______________________________

City, Zip Code: ______________________________

Country: ______________________________

Phone: ______________________________

Please check any industry that applies to you:

□ Automotive
□ Chemical
□ Electrical Machinery
□ Food
□ Instrument and Control
□ Nonelectrical Machinery
□ Petrochemical
□ Pharmaceutical
□ Plastic
□ Pulp and Paper
□ Textiles
□ Transportation
□ Other ______________________________

Remarks Form

Your comments and recommendations will help us to improve the quality and usefulness of our publications. Please take the first available opportunity to fill out this questionnaire and return it to Siemens.

Title of Your Manual: ____

Order No. of Your Manual: ____

Edition: ____

Please give each of the following questions your own personal mark within the range from 1 (very good) to 5 (poor).

1. Do the contents meet your requirements? □
2. Is the information you need easy to find? □
3. Is the text easy to understand? □
4. Does the level of technical detail meet your requirements? □
5. Please rate the quality of the graphics/tables: □

Additional comments:

____

____

____

____

____

____

____

____

____

____

____