

SINUMERIK 840/840C  
SINUMERIK 880/880 GA2  
Computer Link  
SINT Test Software

Planning Guide

09.95 Edition

# **SINUMERIK 840/840C SINUMERIK 880/880 GA2 Computer Link SINT Test Software**

**Planning Guide**

**Manufacturer Documentation**

**Application to:**

<i>Control</i>	<i>Software Version</i>
SINUMERIK 840	from 01
SINUMERIK 840C	from 01
SINUMERIK 880	from 03
SINUMERIK 880 GA2	from 01

**September 1995 Edition**

## Printing history

Brief details of this edition and previous editions are listed below.

The status of each edition is shown by the code in the "Remarks" column.

*Status code in "Remarks" column:*

**A** . . . New documentation.

**B** . . . Unrevised reprint with new Order No..

**C** . . . Revised edition with new status.

If factual changes have been made on the page since the last edition, this is indicated by a new edition coding in the header on that page.

<b>Edition</b>	<b>Order No.</b>	<b>Remarks</b>
05.90	6ZB5 410-0DR02-0BA0	<b>A</b>
03.91	6ZB5 410-0DR02-0AA1	<b>C</b>
09.95	6FC5 197-0AC10-0BP0	<b>C</b>

Other functions not described in this documentation might be executable in the control. This does not, however, represent an obligation to supply such functions with a new control or when servicing.

This publication was produced on the Siemens 5800 Office System.  
Subject to change without prior notice.

The reproduction, transmission or use of this document or its contents is not permitted without express written authority. Offenders will be liable for damages. All rights including rights created by patent grant or registration of a utility model or design, are reserved.

# Preliminary Remarks

**Notes for the reader**

This documentation is intended for installation engineers and users of the SINUMERIK 840/880 computer link who wish to record and analyze message frames at the application level.

There is also a feature that allows you to store the configuration data of the CP in a file on a programmer and to transfer it from there to the system EPROM of the CP.

The description of the SINT software package must be read together with the publication for the 840/880 computer link.

The documentation on the SINUMERIK 840/880 computer link is organized in three parts:

- General documentation
- Configuring
- Installation and Service Documentation.

First-time users of the computer link are advised to refer to the Description "SINUMERIK 840/880 Computer Link" as it includes basic information on the transfer of message frames.

Users who need to record and analyze the message frame transfer during installation or trouble-shooting are advised to refer to the "Description of Message Frames".

Preconditions for the Use of the Test Software  
SINT

---

1

General Description of the Test Software

---

2

General Description of the Operator Interface

---

3

Description of the Screen Forms

---

4

Example of Selective Recording of Message Frames

---

5

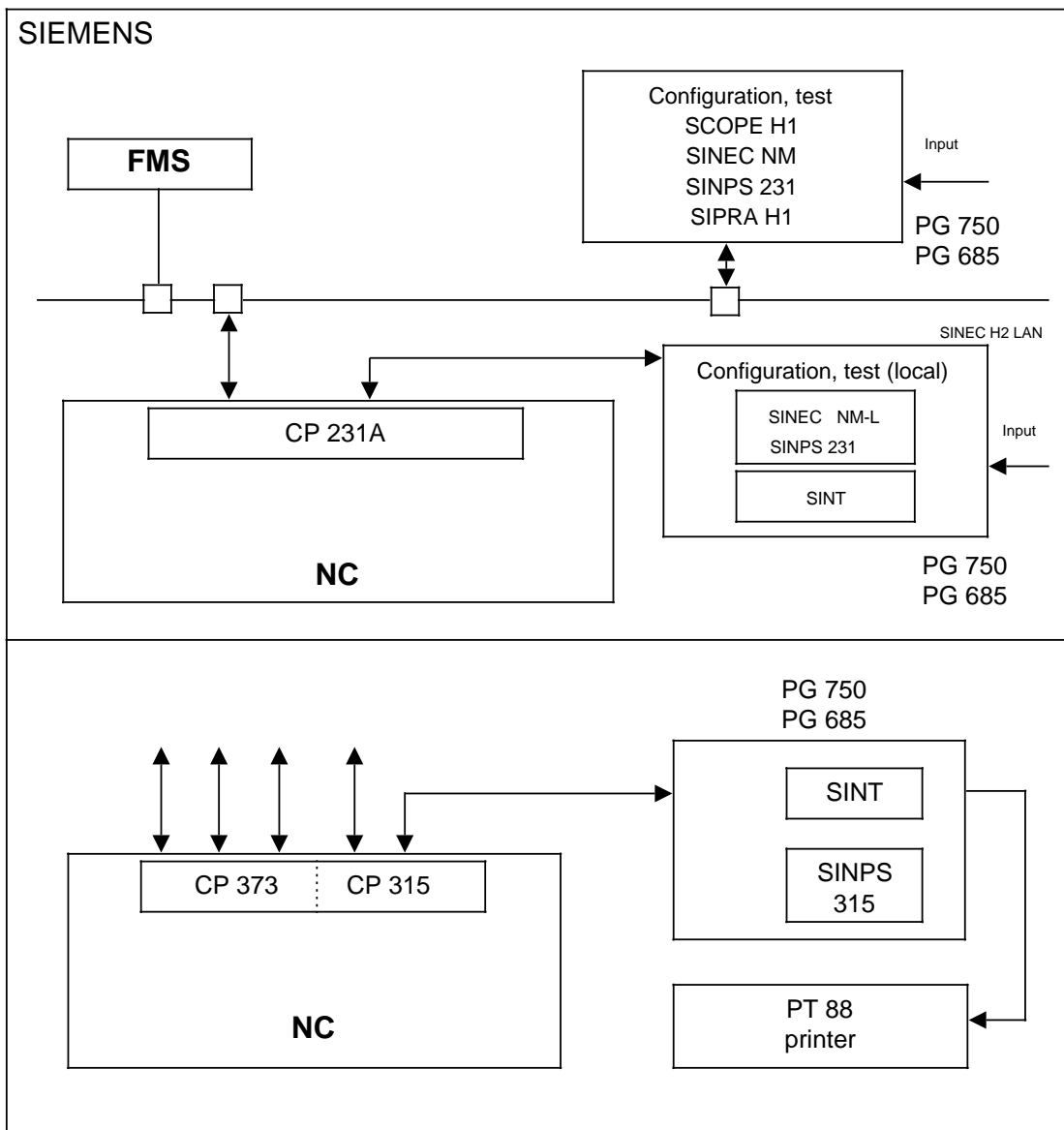
# Contents

	Page
<b>1</b>	<b>Preconditions for the Use of SINT</b> ..... 1-1
1.1	Application ..... 1-1
1.2	Software/hardware requirements ..... 1-2
1.3	Installation of the software ..... 1-2
1.3.1	Copying onto hard disk (PCP/M version) ..... 1-2
1.3.2	Copying to hard disk (MS-DOS version) ..... 1-3
1.4	Overview of versions ..... 1-3
1.5	Interfacing the hardware ..... 1-4
<b>2</b>	<b>General Description of the Test Software</b> ..... 2-1
<b>3</b>	<b>General Description of the Operator Interface</b> ..... 3-1
3.1	Overview of screen forms ..... 3-1
3.2	General structure of screen forms ..... 3-2
3.3	Global function keys ..... 3-3
3.4	Cursor control (PG 685) ..... 3-3
3.5	Cursor control (PG 750) ..... 3-3
<b>4</b>	<b>Description of Screen Forms</b> ..... 4-1
4.1	Screen form 1: Initial form ..... 4-1
4.2	Screen form 2: Selection ..... 4-3
4.3	Screen form 3: Relationships ..... 4-4
4.4	Screen form 4: Triggers ..... 4-6
4.5	Screen form 5: Trace ..... 4-8
4.6	Screen form 6: Addressing data ..... 4-10
4.7	Screen form 7: Useful data ..... 4-12
4.8	Screen form 8/9: Exception trace ..... 4-14
4.9	Screen form 10: Function monitoring ..... 4-15
4.10	Screen form 11: Function and status monitoring ..... 4-16
4.11	Screen form 12: Archiving functions (version 4.0 and higher) ..... 4-18
<b>5</b>	<b>Example of Selective Recording of Message Frames</b> ..... 5-1
5.1	Example of format-related message frame recording ..... 5-1

# 1 Preconditions for the Use of the Test Software SINT

## 1.1 Application

As shown in Fig. 1.1, the test software runs on the PG 685 (PCP/M) or PG 750 (MS-DOS). It can be used in conjunction with the CP 231A and CP 315 interfaces. In terms of hardware, the TTY interface is used for connecting up the CP 231A or CP 315.



*SINUMERIK 840/880 computer link*

*Configuring and testing the CP 315/231A*

## 1.2 Software/hardware requirements

The PCP/M 86 operating system must be installed as basic software on the PG 685 programmer and MS-DOS 3.x on the PG 750 programmer.

For the archiving function (with SINT version 4.0 and higher on the PG 750) you must have at least the following software versions on the CP's:

CP 315	software version 6FX1 841-0BX01	version 3A and higher
CP 231A	software version 6FX1 840-0BX02	version 3A and higher

and the following hardware versions on the CP's:

CP 315	hardware version 6FX1 131-5BA	version 01 and higher
CP 231A	hardware version 6FX1 123-1BC	version 02 and higher

To read and write the EPROM submodule of the CP 315 on the PG750 you need the adapter 6FC9 383-3AA10-0AA0 (order from: AUT WKF-ZSI, Fürth).

## 1.3 Installation of the software

The SINT (PCP/M version) software is supplied on a diskette for the PG 685 I/II. The program cannot be started direct from diskette because the S5WX000H.CMD file, which is necessary for initializing the TTY interface, is loaded on hard disk. The program should therefore be copied onto the hard disk; this also speeds up execution times.

The SINT software (MS-DOS version) is supplied on a diskette for the PG 750. It is possible to start the program directly from the diskette.

### 1.3.1 Copying onto hard disk (PCP/M version)

Before starting the copy function, make sure that there is sufficient memory space on the hard disk (required memory space 400 Kbytes).

The PIP command is used to call the copy function. For example, the software (in User 0 on diskette) is to be installed in User 10 on the hard disk.

```
10 B > PIP B := A : . [ G 0 ]
```

All files in drive 0A are transferred to 10B. When the copy function is finished, the files transferred can be listed using the DIR command.

SINT includes the following files:

SINT	.	CMD
CITYCONV	.	CMD
SINTHELP	.	T
SINTSOFT	.	T
SINTMASK	.	T
SINTMASK	.	TAB
SINTMELD	.	T
SINTMELD	.	TAB
DOSF/RM		
SINTMELD	.	TUK



### 1.3.2 Copying to hard disk (MS-DOS version)

Before copying, you must make sure that there is enough free space on the hard disk (required memory space 500 Kbytes for version 4.0).

You call the copying function with the command COPY. In the following example the software on the diskette is copied onto the hard disk into an existing directory named S5TEST.

```
C > COPY A : . C:\S5TEST
```

The following files belong to SINT:

```
SINT      . EXE
SINF0000 . PDB
SINTHELP . T
SINTSOFT . T
SINTMASK . TAB
SINTMELD . TAB
SINTMELD . TUK
DOSFIRM
```

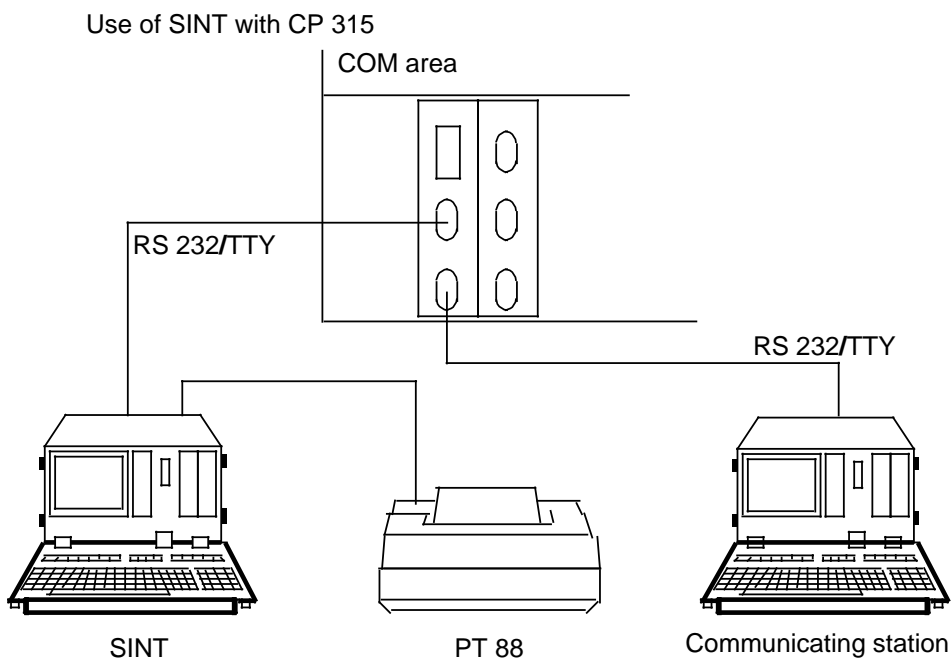
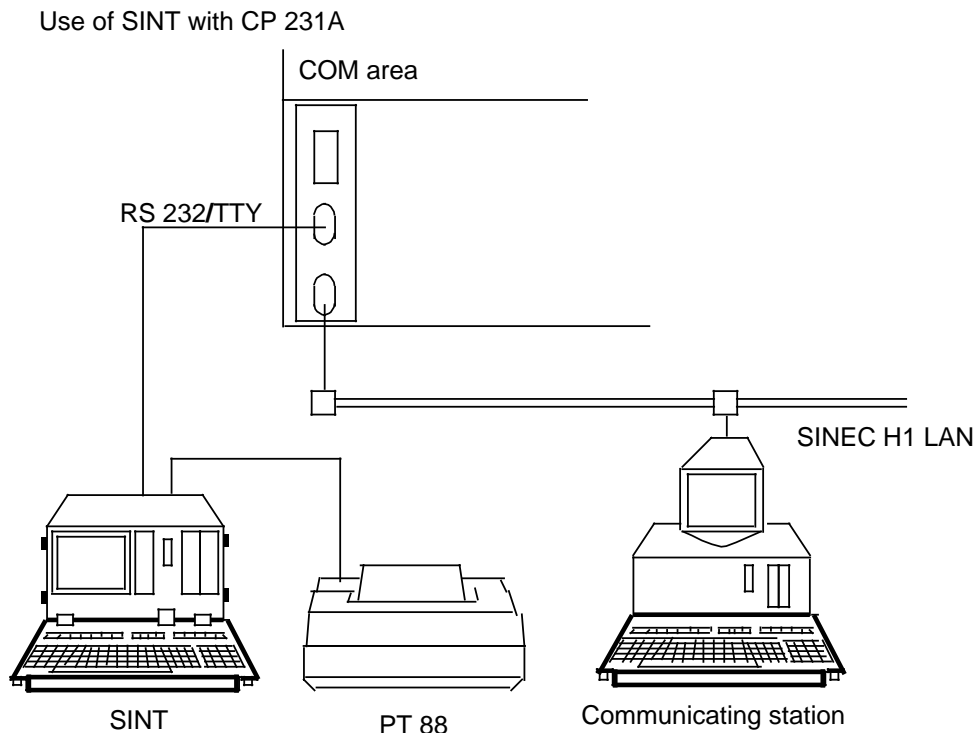
### 1.4 Overview of versions

Version	Order number
SINT for PCP/M on PG 685	6FC9 370-7AC
SINT for MS-DOS on PG 750	6FC9 370-7BC

**Note:**

Please ask your local SIEMENS representative about the current software version.

## 1.5 Interfacing the hardware



*Interfacing the hardware*

**Connecting the PG 685 via the PGAG interface (TTY)**

The transfer parameters are set automatically by the program SINT.

Required cable: 6FC9 340-8G

**Connecting the PG 750 via the COM1 interface (RS232)**

The transfer parameters are set automatically by the program SINT.

Required cable: 6FC9 344-4R

**Note:**

The cable diagrams are contained in the documentation "SINUMERIK System 800, Planning Guide, Universal Interface".

See the SINUMERIK catalog for the cable lengths .

## 2 General Description of the Test Software

The SINT test software is a powerful tool that enables both the installation engineer and the user to start-up, monitor and diagnose message frame transfer at SINUMERIK level.

SINT forms the operator communication and visualization part of the test functions that are installed on the CPs (CP 231A and CP 315).

SINT operates in on-line mode. The connection via the TTY interface on the PG 685 and the RS 232 Interface on the PG 750, however, does not allow real-time acquisition of data (Fig. 2.1).

### The SINT software implements the following test functions:

1. Link status
  2. Error detection on the CP
  3. Listing of the message transfer
1. The status of each link with a logical partner can be scanned (restricted with CP 315).
  2. The module incorporates an 'exception trace' (event buffer) where the various subroutines of the CPs enter warnings and error messages concerning the software modules of the interfaces.
  3. The test software supports the recording of messages on the application level both in the input and output direction, and the display of the net data in a legible form on the programmer.

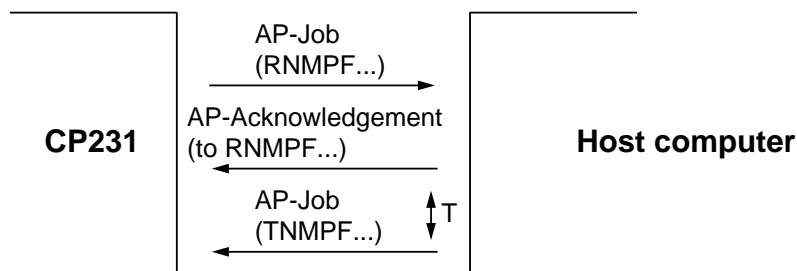
For this purpose, the interface includes two trace buffers that are designed as circular buffers.

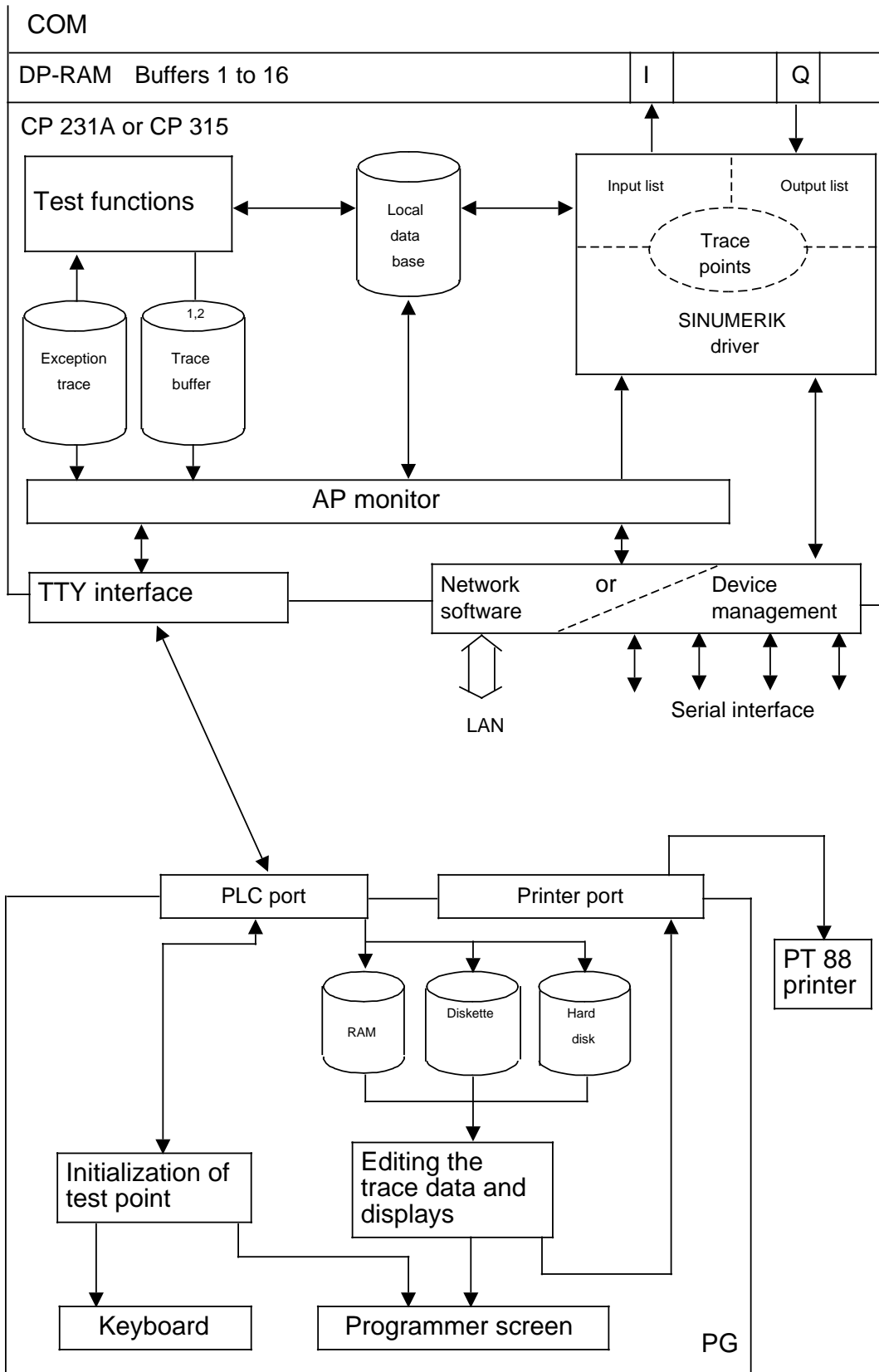
The message frame selection, link selection and various trigger conditions permit selective display of message frames. The user is supported by menu-driven screen forms. The net data can only be displayed in the correct format if a format list is loaded in the CP. The format list must be generated using the SINPSxxx configuring software.

A RAM (approx. 50 Kbytes) is implemented on the programmer for the read-out and display of data. It can be accessed direct from the screen. For larger data volumes, a file can be created on diskette. It is also possible to list the message transfer direct on a printer (via programmer). Data output, i.e. the trace buffers can be disabled by the user program for that purpose (and enabled through SINT on the programmer).

### Note:

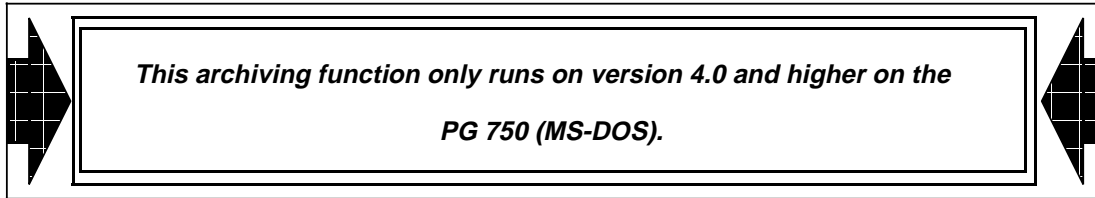
In order to ensure correct data transfer between the CP 231A communications processor and the host computer, the minimum time between two message frames, i.e. between AP acknowledgement and AP new job, should be approx. 30 to 50 ms.





Structure of a SINUMERIK-specific test function

## Archiving function



To establish links between stations (CPs), you must configure parameter sets. This configuring data is backed up by a battery when the power is turned off. However, the data is lost if the battery fails or if the CP is removed. For that reason, the following functions have been implemented:

- Reading out the configuring data of the link via the TTY interface of the CP
- Storing the configuring data on diskette or hard disk
- Transfer of the configuring data to the system EPROM submodule for back-up on replacing the CP. If the back-up battery fails the data stored there is automatically loaded into the main memory and activated on power-on.

The configuring data include the following lists:

- Input list
- Output list
- Format list
- Interface list

This configuring data is stored on the EPROM submodule as a pure data back-up file. If this data is loaded into the RAM after the battery of the EPROM has failed, you cannot change it again from the programmer with the COM software (e.g. to configure another CP with similar data).

### **Note:**

You do not have to use the archiving function.

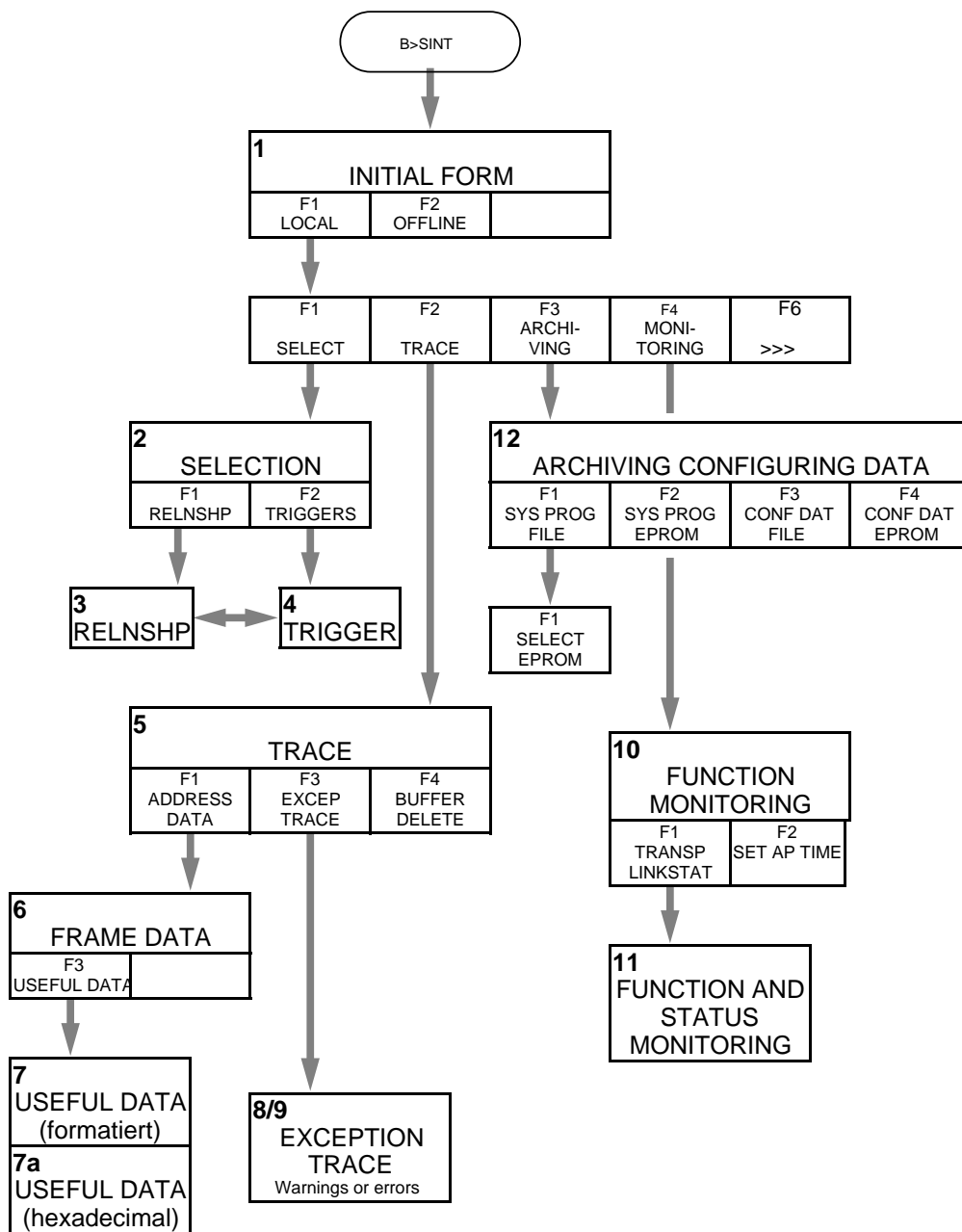
You can continue to store the configuring data only on the RAM (download using SINPS software).

## 3 General Description of the Operator Interface

### 3.1 Overview of screen forms

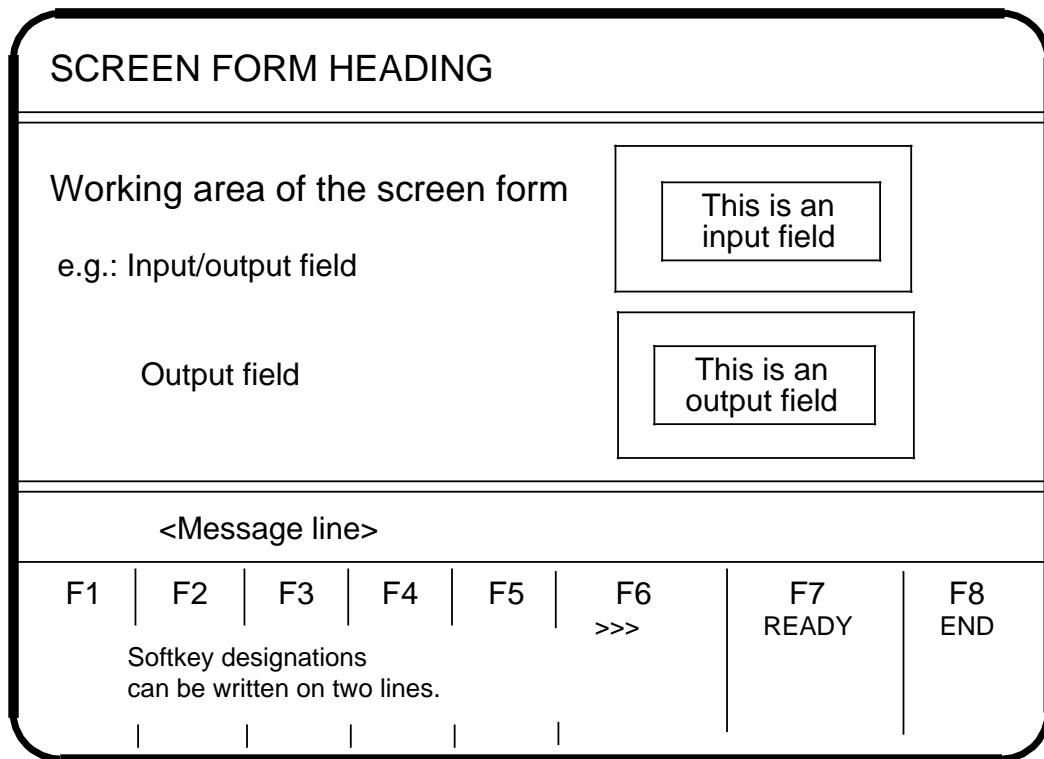
The SINT software features menu-driven screen forms for recording the message transfer and evaluating the entries in the 'exception trace'.

It is also possible to check the parameter assignments of the CPs. The figure below provides an overview of the individual screen forms of SINT and how they can be selected via the menu tree.



Overview of screen forms

### 3.2 General structure of screen forms



The screen form is subdivided into four windows demarcated by horizontal lines.

The first window is a display field containing the screen form header on the left and the program name as a fixed element.

The second window is the working area of the screen form which is subdivided into several input and output fields.

The subdivision into working areas depends on the requirements of the individual screen forms.

The message line includes hints in response to operator entries.

The bottom window shows the current assignments of function keys F1 to F8.



### 3.3 Global function keys

The softkeys are described in detail in the next chapter. The functions of keys F1 to F5 differ according to the screen forms. Keys F6 to F8, however, have global functions and are briefly dealt with in this paragraph. F6 ( >>> ) indicates multiple assignments of keys F1 to F5 (number of functions to be controlled greater than 5).

F7 is used for completing or transferring entries made in the entry line. Softkey F8 is used for going back in the sequence of screen forms.

### 3.4 Cursor control (PG 685)

Data can be selected and entered via the cursor (highlighted field).



The double-line arrow keys can be used to skip from one entry field to another.



Within an entry field, the cursor can be controlled via single-line arrow keys.



If an input field has a default setting, the return key can be used to select the default.

### 3.5 Cursor control (PG 750)

Data can be selected and entered via the cursor (highlighted field).



With these arrow keys you can move the cursor from input field to input field.



Within an input field you can move the cursor with these arrow keys.



If an input field contains a default entry, you can confirm this default with the return key.

## 4 Description of Screen Forms


The various screen forms with the relevant operator entries are described on the following pages.

The following points are dealt with:

- Layout of the screen form
- Functions of the data displayed or data to be entered
- Softkey assignments.

Calling the SINT test software

```
MS-DOS      C > SINT
PCP/M User  B > SINT
```



(if loaded on hard disk)

### 4.1 Screen form 1: Initial form

The initial form is the starting point from where the operator can proceed along the menu tree. See the figure on p. 3-1.

SINTESTTOOLS																							
SINUMERIK -																							
SPECIFIC																							
TEST TOOLS																							
Interface: CP231																							
<table border="1"> <tr> <td>VERSION</td> <td>:</td> <td>4.0</td> <td colspan="5"></td> </tr> <tr> <td>DATE</td> <td>:</td> <td>XX.XX.XX</td> <td colspan="5"></td> </tr> </table>								VERSION	:	4.0						DATE	:	XX.XX.XX					
VERSION	:	4.0																					
DATE	:	XX.XX.XX																					
F1	F2	F3	F4	F5	F6	F7	F8																
LOCAL	OFFLINE					READY	END																
(Format list cannot be interpreted)																							
F1	F2	F3	F4	F5	F6	F7	F8																
SELECT	TRACE	ARCHI- VING	MONI- TORING		>>>	READY	END																

### **Working area:**

In this field, the return key is used to select the required CP (CP 231A or CP 315).

### **Function keys**

#### **1st level**

##### **F1 LOCAL**

F 1 is used to initialize the TTY interface of the CP selected in the working area.

If no format list is loaded on the CP (the entry "okw/o" or "without format list" appears in the "Status" column of form 11 when downloading in SINPS 315 and in the "Status" column of form 3 when downloading in SINPS 231), the message "format list cannot be interpreted" appears in the SINT screen form.

If the interface has been initialized, the program switches over to the second softkey level.

##### **F2 OFFLINE**

The program immediately switches over to the second softkey level instead of scanning the interface.

Data of the internal file can be accessed in the off-line mode.

#### **2nd level**

##### **F1 SELECT**

Call screen form 2: For a preliminary selection of the messages to be recorded.

##### **F2 TRACE**

Call screen form 5: This trace can only be selected after passing the selection branch.

##### **F3 ARCHIVING** (from Version 4.0)

Call screen form 12: Various archiving functions for configuring data.

##### **F6 >>>**

Return to softkey level 1.

## 4.2 Screen form 2: Selection

In this screen form, the operator can select the message frames to be recorded and also at which length they shall be recorded.

SELECTION				SINTESTTOOLS			
Record all messages simultaneously ?				No			
Record only error messages simultaneously ?				No			
Length of messages to be recorded simultaneously ?				010			
Terminate entry by pressing F7!!							
F1	F2	F3	F4	F5	F6	F7	F8
RELNSHP	TRIGGER				>>>	READY	BACK

### Working area

The "Relationships" or "Triggers" screen form can only be called up if "Record all messages simultaneously = No" has been selected.

If all message frames shall be recorded, this test point list can be sent direct to the CP via F7. The message frame length to be recorded is specified in bytes.

### Function keys

#### F1 RELNSHP

Go to screen form 3: Relationships

This screen form shows the links between logic partners.

#### F2 TRIGGER

Go to screen form 4: Trigger

This screen form is used for selecting the trigger conditions (start - stop for message frame recording).

#### F7 READY

Transfer the data selected in the working area.

### 4.3 Screen form 3: Relationships

In this screen form, the operator can select the logic partners for which message frames shall be recorded.

RELATIONSHIPS					SINTESTTOOLS		
No.	Log. partners	Msg.length	Buffer_ID	Activation	Rec. nos.		
1.	...	000	0	Inactive	00000	00000	
2.	...	000	0	Inactive	00000	00000	
3.	...	000	0	Inactive	00000	00000	
4.	...	000	0	Inactive	00000	00000	
5.	...	000	0	Inactive	00000	00000	
6.	...	000	0	Inactive	00000	00000	

F1	F2	F3	F4	F5	F6	F7	F8
TRIGGER		TP_List Initial.	VALUE UPDATE			READY	BACK

#### Working area

**Log. partners :** The user can select individual message frames for a certain link.

**Message length:** Specification of the length of the message frame data to be recorded (maximum length = 236 bytes).

**Buffer\_ID:** The user decides in which of the two trace buffers the message frames shall be recorded (1: buffer 1; 2: buffer 2).

**Activation:** The **return key** can be used to activate or deactivate the buffer.

**Rec.nos.:** Specification of the number of recordings; this value can be updated using F4.

**Function keys****F1 TRIGGER**

Selection of screen form 4: Trigger

**F3 TP\_List Initial.**

Reinitialize test point list.

The old test point list is deleted when operating F 3. A new test point list can be generated using screen forms 3 and 4 (Trigger and Relationships).

**F4 VALUE UPDATE**

The number of recordings of all logic partners is updated.

**F7 READY**

Transfer of the links selected.

## 4.4 Screen form 4: Trigger

In this screen form, the operator can specify the trigger points (start/stop points) of the messages recorded for a particular logic partner.

TRIGGERS				SINTESTTOOLS			
Log. partners :      FLR_							
Buffer_ID                0							
Relationship is        inactive							
Receive messages				Send messages			
Identification		Function no.		Identification		Function no.	
from	to	from	to	from	to	from	to
.....	.....	.....	.....	.....	.....	.....	.....
F1	F2 PAGE+	F3 PAGE-	F4	F5	F6	F7 READY	F8 BACK

### Working area

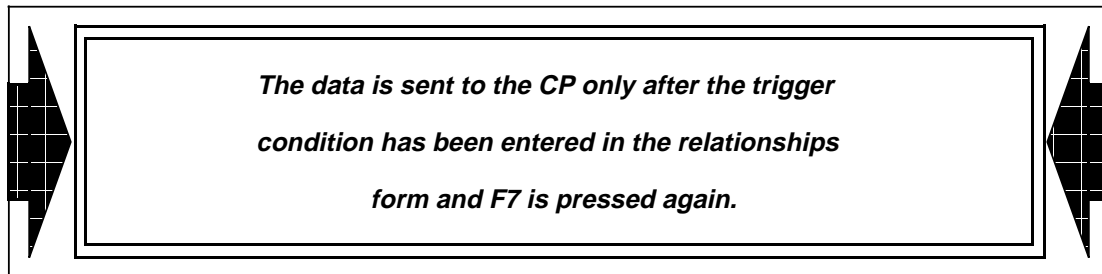
The working area is subdivided into an output field and an input field.

The upper part of the screen form includes the data from the "Relationships" form that is relevant for the selected logic partner.

In the lower part of the screen form, the trigger conditions must be entered. The permissible combinations are shown in the following table.

**Permissible combinations of trigger conditions**

Receive direction				Send direction			
Identifier		Function No.		Identifier		Function No.	
from	to	from	to	from	to	from	to
X	X						
X			X				
X					X		
	X	X					X
	X			X			
		X				X	
		X	X				X
			X				
		X					X
				X	X		
				X			X
					X	X	
						X	X

**Function keys****F2 PAGE +**

Select the next logic link in the Relationships table.

**F3 PAGE -**

Select the previous logic link.

**F7 READY**

Transfer trigger conditions to trigger memory.



### 4.5 Screen form 5: Trace

The CP 231A or CP 315 interfaces incorporate two trace buffers for recording the message transfer and an 'exception trace' where warnings and errors concerning the interface are entered (for details on the 'exception trace' refer to Chapter 4.8).

In the trace form, the operator can decide which buffers shall be active and whether the data is to be listed on a printer (PT), displayed on the programmer or stored in a file.

TRACE				SINTESTTOOLS			
INPUT UNIT:  Filename:     .....  Exception trace    Do not read  Trace memory Buffer 1        Buffer 2  Do not read        Do not read  Locked             Locked				OUTPUT UNIT:  Filename :        .....  Printer            Inactive  Screen            Active			
F1 ADDRESS DATA	F2	F3 EXCEPTION TRACE	F4 BUFFER DELETE	F5	F6	F7 READY	F8 BACK

#### Working area

The working area is subdivided into the input area "Trace selection" (left-hand side) and the output unit.



The double-line arrow keys can be used to move from the input unit to the output unit, and vice versa.



In the output unit, the name of the file where the buffer contents are to be stored must be entered next to "File name" (terminate entry with return key).

If the data of this file (hard disk) is to be displayed or printed out, the file name must be specified in the input unit.

Large volumes of data can therefore either be printed out or transferred to the hard disk of the programmer.

It is also possible to stop data output via the PLC (DB 101). This can be utilized for event-controlled recording of large volumes of data.

Only one input and one output unit can be active simultaneously.

### Function keys

#### **F1 ADDRESS DATA**

Go to screen form 6: addressing data if entries have been made in the trace buffer that were read out with F7.

#### **F3 EXCEPTION TRACE**

Call screen form 8/9: Exception trace - if entries have been made in the trace buffer that were read out with F7.

#### **F4 BUFFER DELETE**

The buffers can only be deleted if they have been declared inactive.

#### **F7 READY**

Transfer the data selected in the working area.


When you press F7 again, the data is transferred from the CP to the programmer, the dialog text "Data being read in, abort with F7" indicates that entries are in the trace buffer.

## 4.6 Screen form 6: Frame data

This screen form provides an overview of the message frame data, excluding the useful data.

ADDRESSING DATA						SINTESTTOOLS																	
from input unit: Trace buffer 1																							
PNR.	DATE	TIME	LOG.PART.	IDENT.	FCTNO.	ERRNO.	COMM.																
004	01.01	00:04:22:750	DRU2 CP<-	T TG F **	00021	00100	FORM 2																
005	01.01	00:06:21:760	DRU2 CP<-	T TG **	00036	00000	XXX																
006	01.01	00:06:33:560	DRU2 CP <-	T TG **	00036	00000	XXX																
007	01.01	00:06:42:490	DRU2 CP<-	BBBBBB **	00100	00000	FORM 1																
008	01.01	00:06:42:550	DRU2 CP->	BBBBBF **	00100	00008	NO FORMAT DEF																
009	01.01	00:06:51:790	DRU2 CP<-	BBBBBB **	00100	00000	FORM 1																
<table border="1"> <tr> <td>F1 LINE+</td> <td>F2 LINE-</td> <td>F3 USEFUL DATA</td> <td>F4 BUFFER REREAD</td> <td>F5</td> <td>F6 &gt;&gt;&gt;</td> <td>F7 READY</td> <td>F8 BACK</td> </tr> <tr> <td>F1 PAGE+</td> <td>F2 PAGE-</td> <td colspan="3"></td> <td>F6 &gt;&gt;&gt;</td> <td colspan="2"></td> </tr> </table>								F1 LINE+	F2 LINE-	F3 USEFUL DATA	F4 BUFFER REREAD	F5	F6 >>>	F7 READY	F8 BACK	F1 PAGE+	F2 PAGE-				F6 >>>		
F1 LINE+	F2 LINE-	F3 USEFUL DATA	F4 BUFFER REREAD	F5	F6 >>>	F7 READY	F8 BACK																
F1 PAGE+	F2 PAGE-				F6 >>>																		

### Working area

The entries in the working area are explained in the header. The cursor is used to select a message frame; its useful data can be called up by pressing F 3. When using the double-line arrow key, the cursor  must first be moved to the right before going downwards.

### Function keys

#### F1 LINE +

The entries are shifted upwards by one line.

#### F2 LINE -

The entries are shifted downwards by one line.

**F3 USEFUL DATA**

Calling screen form 7 or 7a.

If a format list is loaded, the useful data is displayed in the correct format in screen form 7. If no format list is available, the program immediately goes to screen form 7a where the useful data is displayed as hexadecimal numbers or in ASCII code.

**F4 BUFFER REREAD**

The internal buffer on the interface is read again.

**F6** Switches over to the second softkey level.

**2nd softkey level****F1 PAGE +**

Pages forward if the entries exceed the length of one page.

**F2 PAGE -**

Pages backwards, page by page.

### 4.7 Screen form 7: Useful data

This screen form shows all the data related to a message frame. If the test software finds a complete format list, the useful data is displayed in plaintext in accordance with the specified format and comments; otherwise, it will switch to screen form 7a.

USEFUL DATA		from input unit:			Trace buffer 1		
PNR.	DATE TIME	LOG.PART.	IDENT.	FCTNO.	ERRNO.	COMM	
139	12.08. 14:30:10:050	FLR CP<-	T_WB	026	0000	WZ_ DAT	
Magazine no.	1	Geometry 1	1				
Locator no.	1	Wear 2	1				
Identifier	0000	Wear 3	0000				
P no.	3841	Wear 1	3841				
Tool type	1	Tool life	1				
Locations, front	3	Reserve 1	3				
Geometry 1	100000560	Reserve 2	100000560				
<MESSAGE LINE>							
F1 Page+	F2 Page-	F3 Line+	F4 Line-	F5	F6 >>>	F7 READY	F8 BACK

#### Working area

The working area is subdivided into two columns of equal size. The comments are shown on the left whereas the formatted values of the message frame appear on the right.

## Screen form 7a: Useful data

This screen form displays the message frame data without format list. The useful data is represented in hexadecimal form.

USEFUL DATA										SINTESTTOOLS									
HEXADECIMAL DISPLAY																			
from input unit: Trace buffer 1																			
PNR.	DATE	TIME	LOG.PART		IDENT.	FCTNO.	ERRNO.	EXT	COMM.										
001	01.01	00:02:40:130	DRU1	CP<-	E 000	XXXXX	00000		NO SYS800 MESS.										
54	20	54	47	20	20	00	FF	01	02	03	04	05	06	T TG					
07	08																		
<table border="1"> <tr> <td>F1 LINE+</td> <td>F2 LINE-</td> <td>F3 PAGE+</td> <td>F4 PAGE-</td> <td>F5</td> <td>F6</td> <td>F7 READY</td> <td>F8 BACK</td> </tr> </table>												F1 LINE+	F2 LINE-	F3 PAGE+	F4 PAGE-	F5	F6	F7 READY	F8 BACK
F1 LINE+	F2 LINE-	F3 PAGE+	F4 PAGE-	F5	F6	F7 READY	F8 BACK												

### Working area

The useful data is represented in hexadecimal form.

### Function keys

#### F1/F2 LINE + / LINE -

Move up or down one line.

#### F3/F4 PAGE + / PAGE -

Scrolls forwards or backwards one page if the entries exceed the length of one page.

## 4.8 Screen form 8/9: Exception trace

### Screen form for warnings and errors

The software of the interface includes an 'exception trace' that enables the operator to track down faults and exceptions in the internal program execution. Depending on the event, either warnings or error forms are used. Both forms have the same structure.

EXCEPTION TRACE						SINTESTTOOLS	
WARNINGS							
PNR	DATE	TIME	SUBSYST.	ERRNO.	OBJECT	ERR.CLASS	ERR.CODE
00000	01.01	00:07:45:930	SIN - Tr	156	A DRU1	CONFIG	# 1
00001	01.01	00:07:46:310	SIN - Tr	156	A DRU1	CONFIG	# 1

F1 LINE+	F2 LINE-	F3 PUFFER NEULESEN	F4	F5	F6 >>>	F7	F8 BACK
-------------	-------------	--------------------------	----	----	-----------	----	------------

### Working area

Entries from the 'exception trace' are displayed in this screen form.

**SUBSYST.:** Abbreviations of subsystems that can be entered in the trace, such as AP monitor, device management, SINUMERIK drivers, etc.

All other entries, regarding error number, object identifier, error class and error code, can be used by the operator for detecting the causes of internal errors in the interface.

**ERROR NO.:** Error number that identifies errors occurring in the internal program execution of the interface.

**OBJECT:** Object processed by the subsystem (e.g. link/channel).

**ERROR CLASS:** Coding and categorizing the errors that have occurred according to error classes.

**ERROR CODE:** Detailed identification of the error with regard to the error class.

## 4.9 Screen form 10: Function monitoring

This screen form shows the current status of the interface module. You can also use it to set the current date and time on the module.

FUNCTION MONITORING				SINTESTTOOLS			
SCP	Status:	# 0400	Run				
	Cause:	0000					
LDS	Status:	# 0003	Consistent				
	Cause:	0000	OK				
Trace	Status:	# 01	Trace buffer 1 active				
AP time:	-	-	h m s	000	ms		
no AP time set							
F1	F2	F3	F4	F5	F6	F7	F8
TRANSP LINKSTAT	SET AP TIME						BACK

### Working area

- SCP:** Specifies whether the module is in the stop state (also specifying the cause) or in the RUN mode.
- LDS:** Status of the local data base on the module: "Consistent" signifies that the result of the validity check carried out on the data is o.k.
- Trace:** Status of the two trace buffers: either both are disabled or one of them is ready to record simultaneously.
- AP time:** Entry of date and time, transfer through F2, set AP time.  
**Specify day minus 1** since the program adds 1 during transfer.

### Function keys

- F1: TRANSP LINKSTAT**  
Calls screen form 11: Function and status monitoring in which the status of the transport link or the logic partner are specified.
- F2: SET AP TIME**  
Transfer AP time.



### 4.10 Screen form 11: Function and status monitoring

The statuses of the links between logic partners and of the logic partners themselves are displayed in this screen form. The settings of the send and receive credits are also shown.

FUNCTION AND STATUS MONITORING				SINTESTTOOLS			
NAME	LPAR STATUS	LINK STATUS	RECEIVE CREDIT	SEND CREDIT			
NCFLR SYSTEM		WAIT_STRT_UP	001T	STR	001T	STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	
		WAIT_STRT_UP		STR		T STR	

KEY DISABLED							
F1 PAGE+	F2 PAGE-	F3	F4	F5	F6	F7 READY	F8 BACK

#### Working area

**NAME:** Name of the logic \_link.

**LPAR STATUS:** Status of the logic partner

Possible values:

- **SYSTEM**  
Static logic partner in a system link (for CP 315)
- **STATIC**  
Static logic partner in a standard link, which is activated in the RUN state.
- **DYN - ATTACH**  
Dynamic logic partner; the logic link can be activated using the 'Attach' command.

**LINK STATUS :** The status of the link is displayed.

Possible values:

- **WAIT-STRT-UP**  
No link established.
- **PUT - UP**  
The link is being established.
- **RECOVERY**  
The link is currently being reestablished after a disruption.
- **READY**  
The link is ready.

### Function keys

**F1/F2**    **PAGE + / PAGE -**

Page forwards or backwards.

## 4.11 Screen form 12: Archiving functions (version 4.0 and higher)

With the archiving function, you can store the configuration data of the links on the system EPROM submodule of the CP.

You can only store the configuration data on the EPROM submodule if the system program is already there.

It is best to read out the system program at the start and store it as a file. If you later have to delete the EPROM to change the configuration data, you can blow the system program onto the beginning of the EPROM again. This is necessary because it is not possible to delete parts of EPROMs.

With screen form 12, you can select various archiving functions required to store the configuration data on the system EPROM.

ARCHIVING				SINTESTTOOLS			
INPUT UNIT:				OUTPUT UNIT:			
F1 SYS PROG FILE	F2 SYS PROG EPROM	F3 CONF DAT FILE	F4 CONF DAT EPROM	F5	F6	F7	F8 BACK

Different input and output units are active depending on the archiving function selected.

The possible inputs are described in the explanations of each subscreen.

### Function keys

<b>SYS.PROG FILE</b>	Storing the system program from the EPROM submodule onto diskette/hard disk
<b>SYS.PROG EPROM</b>	Blowing the system program from the diskette/hard disk onto the EPROM submodule
<b>CONF DAT FILE</b>	Storing the configuration data from the EPROM submodule onto diskette/hard disk
<b>CONF DAT EPROM</b>	Blowing the configuration data from the diskette/hard disk onto the EPROM submodule

### Note:

The configuration data and the system data are transferred onto the EPROM submodule in two passes. The first pass checks the data consistency and the second transfers the data.

During the **archiving** function the following error messages might be output:

### FILE errors

- Error on opening a file (file non-existent, not found)
- Error on reading a file! Repeat? Y/N
- Error on writing a file
- Error on seek call (file destroyed or internal error)
- Error on closing a file
- File already exists! Overwrite? Y/N
- Block is empty/unformatted (database is not in CP)
- Block is inconsistent (configuration data are destroyed, parity bit incorrect)
- Output file: writing error (diskette full, diskette write-protected etc.)
- Input file cannot be interpreted (file length does not agree, probably error on reading in the database from the interface module)
- Memory requirement rejected (not enough usable main memory available)
- SCP error (data is destroyed, parity error, the exact cause of error can be determined the softkey function "monitoring" in the main menu)
- Function only possible locally (on-line) (configuration data can only be read out of the CP when the programmer is in on-line mode).

### Firmware or device error

- Incorrect command (internal error, firmware defective)
- Device does not respond (EPROM interface module defective)
- File is larger than EPROM (EPROM capacity insufficient for file)
- Abort by user (function aborted by software)
- Error from device (EPROM interface module defective)
- Programming error socket 1 (EPROM defective)
- Programming error socket 2 (EPROM defective)
- Programming error socket 3 (EPROM defective)
- Programming error socket 4 (EPROM defective)

## 4.11 Screen form 12: Archiving functions (version 4.0 and higher)

- EPROM 1 not deleted
- EPROM 2 not deleted
- EPROM 3 not deleted
- EPROM 4 not deleted
- Voltages not OK (internal hardware error)
- Wrong firmware version (wrong driver software loaded, re-install)
- Overcurrent Ipp (internal hardware error)
- Unknown type code (an unknown algorithm type was input)
- Unknown command (an unknown algorithm type was input)
- Vcc incorrect (internal hardware error)
- Odd number of data (final address for programming is an odd address)
- Odd starting address (initial address for programming is an odd address)
- Overcurrent Icc (internal hardware error)
- Incorrect submodule plugged in (an incorrect submodule type was specified)
- No submodule plugged in
- Job version incomprehensible (internal firmware error)
- Adapter not plugged in (you need an adapter to program this submodule)
- Adapter unnecessarily plugged in (you do not need an adapter to program this submodule but an adapter is plugged in)
- Wrong type of EPROM in the submodule (this type of EPROM cannot be used in this submodule)
- Error on opening DOSfirm (software not installed correctly)
- Firmware runtime error (internal firmware error)
- PROGAS not available (EPROM interface module defective or not present)
- PRINTER BUSY, PLEASE WAIT (a "\*" was input as a filename).

## Screen form 12a: Archiving the system program to file

You must copy the system program into a file from the EPROM submodule before blowing the configuration data onto the EPROM submodule for the first time. To do this you first have to slot an EPROM submodule into the EPROM slot of the programmer. You must remove the CP module during this time because configuration data would then no longer be battery-backed.

ARCHIVING				SINTESTTOOLS			
COPY SYSTEM PROGRAM TO FILE							
INPUT UNIT: EPROM ORDER NO: XXXXXXX-XXXXX TYPE: XXXXXXXXXXXXX				OUTPUT UNIT: FILE			
FILE: L:\path...\name Copying... (error messages)							
F1 SELECT EPROM	F2	F3	F4	F5	F6 ABORT	F7 READY	F8 BACK

### Working area

You have to select the order No. and the type of the submodule in the submenu SELECT EPROM (function F1).

The name of the output file you enter has to be a DOS filename, optionally with the drive and the path. The extension of the filename is given by the system (**.2SP** for CP 231A and **.3SP** for CP 315).

The active copying function is displayed blinking in the status message **copying...** .

### Function keys

- F1** Selection of the EPROM submodule (see screen form 12e)
- F6** Abort the copying function after it was started
- F7** Start the copying function
- F8** Return to previous menu.

## Screen form 12b: Blowing the system program onto the EPROM

You must store the system program on an empty system EPROM submodule before you put the configuration data onto it. To do this, you first have to slot an EPROM submodule into the EPROM slot of the programmer.

ARCHIVING				SINTESTTOOLS			
COPY SYSTEM PROGRAM TO EPROM							
INPUT UNIT: FILE				OUTPUT UNIT: EPROM			
				ORDER NO: XXXXXXXX-XXXXXX			
				TYPE: XXXXXXXXXXXXXXXX			
FILE: L:\path...\name							
Copying... (error messages)							
F1 SELECT EPROM	F2	F3	F4	F5	F6 ABORT	F7 READY	F8 BACK

### Working area

You have to select the order No. and the type of the submodule in the submenu SELECT EPROM (function F1).

The name of the output file you enter has to be a DOS filename, optionally with the drive and the path. You must not enter the extension of the filename but it is checked by the system (.2SP for CP 231A and .3SP for CP 315).

The active copying function is displayed blinking in the status message **copying...** .

This function runs in two passes. The first pass is a blank test of the EPROM submodule and the second transfers the system program to the EPROM submodule.

### Function keys

- F1** Selection of the EPROM submodule (see screen form 12e)
- F6** Abort the copying function after it was started
- F7** Start the copying function
- F8** Return to previous menu.

## Screen form 12c: Archiving the configuration data in a file

Before blowing the configuration data onto the EPROM submodule, you must first copy it from the CP (for the cable connection see Section 1.5) onto a file on the diskette or hard disk.

ARCHIVING				SINTESTTOOLS			
COPY CONFIGURATION DATA TO FILE							
INPUT UNIT: LOCAL DATABASE				OUP TUT UNIT: FILE			
FILE: L:\path...\name Copying block X (error messages)							
F1 SELECT EPROM	F2	F3	F4	F5	F6 ABORT	F7 READY	F8 BACK

### Working area

The name of the output file you enter has to be a DOS filename, optionally with the drive and the path. The extension of the filename is given by the system (**.2SP** for CP 231A and **.3SP** for CP 315).

The active copying function is displayed blinking in the status message **copying block X**.

### Function keys

- F1** Selection of the EPROM submodule (see screen form 12e) not required here
- F6** Abort the copying function after it was started
- F7** Start the copying function
- F8** Return to previous menu.



### Screen form 12d: Blow configuration data onto an EPROM submodule

Before you transfer the configuring data (database) to the EPROM submodule, the system program must already be there.

For this function an EPROM submodule must be slotted into the EPROM slot of the programmer.

ARCHIVING		SINTESTTOOLS					
COPY CONFIGURATION DATA TO EPROM							
INPUT UNIT: FILE				OUTPUT UNIT: EPROM ORDER NO: XXXXXXX-XXXXX TYPE: XXXXXXXXXXXXX			
Caution: Before you program the database onto the EPROM submodule, make sure that the system program is backed up!							
FILE: L:\path...\name Copying block X (error messages)							
F1 SELECT EPROM	F2	F3	F4	F5	F6 ABORT	F7 READY	F8 BACK

#### Working area

You have to select the order No. and the type of the submodule in the submenu SELECT EPROM (function F1).

The name of the output file you enter has to be a DOS filename, optionally with the drive and the path. You must not enter the extension of the filename but it is checked by the system (.2SP for CP 231A and .3SP for CP 315).

The active copying function is displayed blinking in the status message **copying block X** .

This function runs in two passes. The first pass is a blank test of the EPROM submodule and the second transfers the system program to the EPROM submodule.

#### Function keys

- F1** Selection of the EPROM submodule (see screen form 12e)
- F6** Abort the copying function after it was started
- F7** Start the copying function
- F8** Return to previous menu.

## Screen form 12e: Selection of the EPROM submodule

Before you can read in the system program from the submodule or blow data files onto the EPROM submodule you must first specify the type of EPROM submodule.

ARCHIVING				SINTESTTOOLS			
SELECT EPROM							
ORDER NO:		ADAPTER		ALGORITHM TYPE:			
1: 6XF1126-0BC00		N		1: NMC27C256Q250	10: D27C512D-20		
2: 6FX1126-0BB00		N		2: AM27C256-250DC	11: AM27C512-250DC		
3: 6FX1130-5BB00		N		3: D27C256AD-20	12: AM27C512DC		
4: 6FX1130-5BA00		J		4: TMS27C256-25JL	13: NMC27C512AQ250		
5: 6FX1128-4BB00		J		5: 27C256-25FA	14: AM27C512-255DC		
6: 6FX1128-4BA00		J		6: TMS27C256-2JL	15: TMS27C512JL		
7: 6FX1122-6CB00		N		7: TMS27C256-25JL	16: D27C2001		
8: 6FX1122-6CA00		N		8: MBM27512-25	17: 27C020		
				9: TMS27C512-25JL	18:		
Number: X				Number: X			
Input number (change input field with RETURN)							
F1	F2	F3	F4	F5	F6	F7	F8
						READY	BACVK

### Working area

You select the order No. of the submodule and the type of algorithm of the EPROMs used by inputting the appropriate number. The order number is written on the submodule, the type of algorithm is to be found on the label of the EPROM used. Press the return key to swap between the input fields for the order No. and the algorithm type.

If a PG 750 adapter is required to read and write the EPROM (see Section 1.2), this is indicated by a **Y**.

### Function keys

- F7** Return to previous menu with retaining the submodule selection
- F8** Return to previous menu without retaining the submodule selection.

## Overview of the submodule and EPROM types supported

The following table shows the possible assignments of EPROMs to submodule types

### Submodule 1: 6FX1126-0BC00

1: NMC27C256Q250	NSC
2: AM27C256-250DC	AMD
3: D27C256AD-20	NEC
4: TMS27C256-25JL	Texas Instruments
5: 27C256-25FA	ValvoSignetico
6: TMS27C256-2JL	Texas Instruments
7: TMS27C256-25JC	Texas Instruments

### Submodule 2: 6FX1126-0BB00

1: NMC27C256Q250	NSC
2: AM27C256-250DC	AMD
3: D27C256AD-20	NEC
4: TMS27C256-25JL	Texas Instruments
5: 27C256-25FA	ValvoSignetico
6: TMS27C256-2JL	Texas Instruments
7: TMS27C256-25JC	Texas Instruments

### Submodule 3: 6FX1130-5BB00

1: NMC27C256Q250	NSC
2: AM27C256-250DC	AMD
3: D27C256AD-20	NEC
4: TMS27C256-25JL	Texas Instruments
5: 27C256-25FA	ValvoSignetico
6: TMS27C256-2JL	Texas Instruments
7: TMS27C256-25JC	Texas Instruments

### Submodule 4: 6FX1130-5BA00

8: MBM27C512-25	Fujitsu
9: TMS27C512-25JL	Texas Instruments
10: D27C512D-20	NEC
11: AM27C512-250DC	AMD
12: AM27C512DC	AMD
13: NMC27C512AQ250	NSC
14: AM27C512-255DC	AMD
15: TMS27C512JL	Texas Instruments

### Submodule 5: 6FX1128-4BB00

8: MBM27C512-25	Fujitsu
9: TMS27C512-25JL	Texas Instruments
10: D27C512D-20	NEC
11: AM27C512-250DC	AMD
12: AM27C512DC	AMD
13: NMC27C512AQ250	NSC
14: AM27C512-255DC	AMD
15: TMS27C512JL	Texas Instruments

**Submodule 6: 6FX1128-4BA00**

8: MBM27C512-25	Fujitsu
9: TMS27C512-25JL	Texas Instruments
10: D27C512D-20	NEC
11: AM27C512-250DC	AMD
12: AM27C512DC	AMD
13: NMC27C512AQ250	NSC
14: AM27C512-255DC	AMD
15: TMS27C512JL	Texas Instruments

**Submodule 7: 6FX1122-6CB00**

16: D27C2001	NEC
--------------	-----

**Submodule 8: 6FX1122-6CA00**

17: 27C020	INTEL
------------	-------

**Note:**

- Submodule type 6 (6FX1128-4BA version 0.0 and higher) is used with the CP 315 (submodule adapter required, see Section 1.2).
- Submodule type 7 (6FX1122-6CB version 0.0 and higher) is used with the CP 231A.

## 5 Example of Selective Recording of Message Frames

### Task:

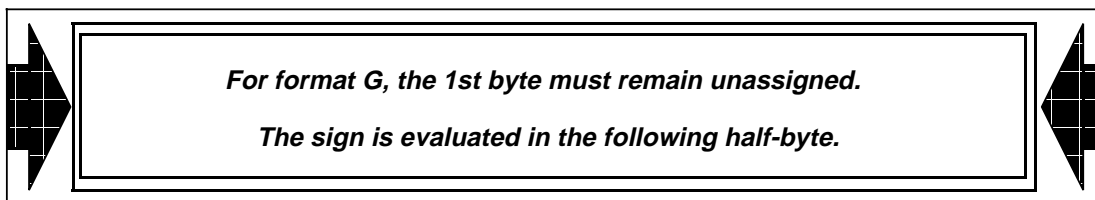
The message frames of the logic partner ???? are to be recorded in the input direction. Recording is initiated by the message frame with function number 14 (mode = "changeover") and terminated by function number 10 (T OK). The permissible program length shall be 200 bytes. The message frames are to be entered in buffer 1.

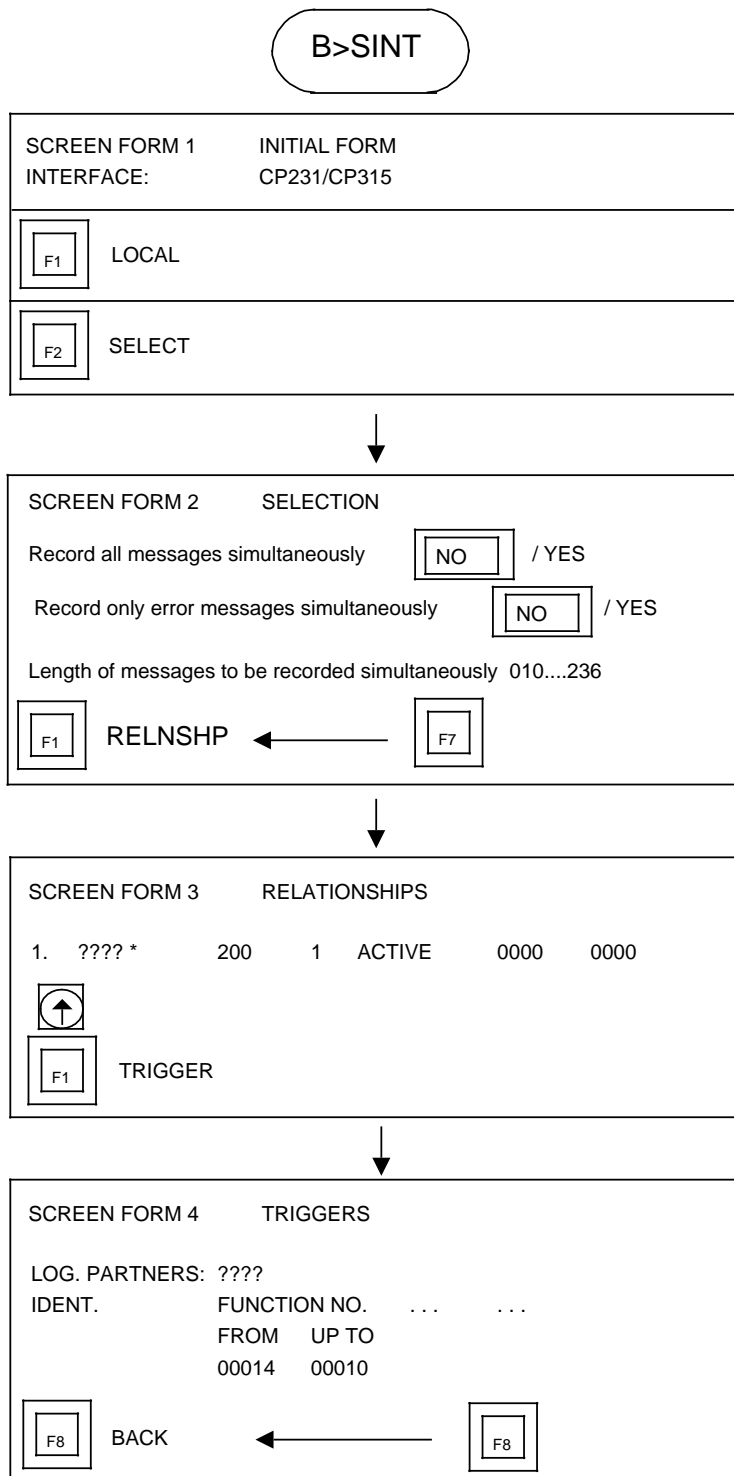
### 5.1 Example of format-related message frame recording

A format list for the FOEI user message frame (Function No. 101) is shown on page 5-5. It has been generated with the SINPS 315 configuring software (see Figure 5.1, above).

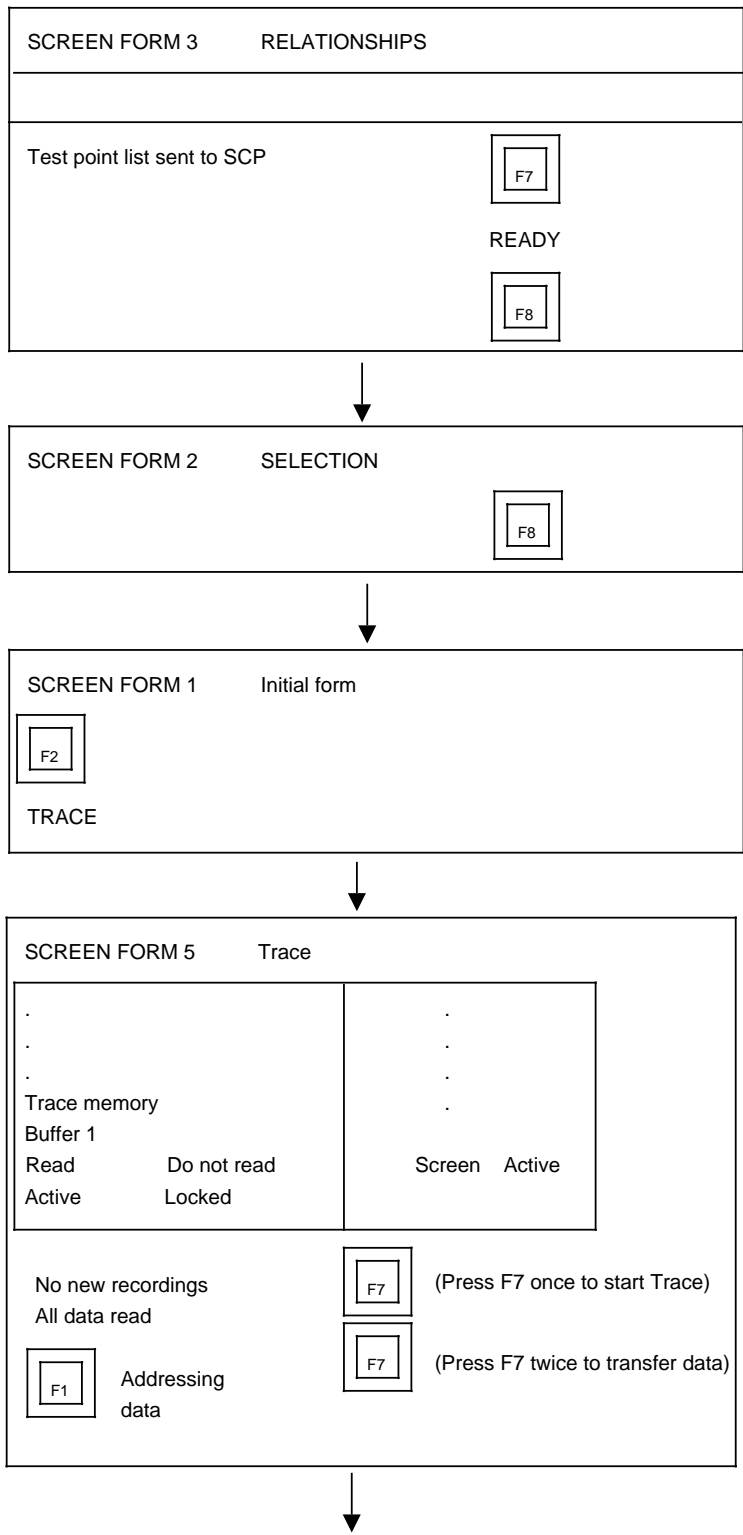
The message frame itself was generated using FOXPG-S and was then sent to the CP 315 (see Figure 5.1, centre).

In the lower part of Figure 5.1, a recorded message frame is shown with the various number formats and associated comments.







? = ASCII characters for logic partners of addressing list



↓

Screen form 6      Addressing data  
 of input unit:      Trace buffer 1

FNR.	DATE	TIME	LOG.PART.	IDENT.	FCT.NO.	ERRNO.	COMM.
			→ Message frame selection for display of useful data →				
							
000	0.0.	.00:15	FLR_CP <-	C MO . . .	014	00000	----
001	0.0.	.00:15	FLR_CP <-	PTMPF	029	00000	----
002	0.0.	.00:15	FLR_CP <-	TNMPF	030	00000	----
003	0.0.	.00:15	FLR_CP <-	T OK	010	00000	----

 USEFUL DATA

↓

↓

Screen form 7      Useful data

PNR.	DATE	TIME	LOG.PART.	IDENT.	FCTNO.	ERRNO.	COMM.
001	01.01.86	00:09	FLR_CP <-	PTMPF	029	00000	FILETRANS
"Useful data - formatted"							
from MPF		1111					
up to MPF		1111					



SINPS 315

---

FORMAT LIST

---

	Comment	Format of useful data	Repetition factor
--	---------	-----------------------	-------------------

---

Format name :           FORMAT LIST

---

0	WORKPIECE	C	5
	NUMBER	F	1
	TOOL	H	1
	CASSETTE	M	1
	BCD without	O	1
	BCD with	G	1

---

FOX PG - S

	AS 512 header	Length in bytes	Identifier	Error number
<b>F1 :</b>	<41X44X22X01X00X12X01X01X00X22X46X4FX45X49X20X20X20X00X00>			
	C5           F1 H1       M1			
	<4DX4FX54X4FX52X00X0AX00X0AX00X0A>			
	01           G1			Filler byte for even number
	<00X01X23X45X67X89X00X00X78X9EX01X09X20>			
	Sign   0 =+ / 1 = -			
	according to SINUMERIK format (function macros 3 to 6)			

---

SINT

TRACE DATA FROM TRACE BUFFER 1

NET DATA:

PNR.	DATE	TIME	LOG.PART.	IDENT.	FCTNO.	ERRNO.	COMM.
001	10.03.	10:20:05:300	FLR CP <-	FDEI	101	00000	FORMAT LIST
			WORKPIECE	MOTOR			
			NUMBER	10			
			TOOL	000A			
			CASSETTE	0000000000001010			
			BCD without	000123456789			
			BCD with	+0789.0109			

Fig. 5.1   Format list, message frame and addressing data referring to Example 5.1